# DATA-DRIVEN METHODOLOGIES FOR DECISION MAKING IN ENGINEERING DESIGN

BY

DEDY SURYADI

DISSERTATION

Submitted in partial fulfillment of the requirements
for the degree of Doctor of Philosophy in Industrial and Enterprise Systems Engineering
in the Graduate College of the
University of Illinois at Urbana-Champaign, 2019

Urbana, Illinois

Doctoral Committee:

Professor Harrison M. Kim, Chair
Professor Deborah Thurston
Associate Professor Julia Hockenmaier
Associate Professor Pingfeng Wang

# Abstract

In the product development process, customer needs are essential to develop the product concepts. These concepts are crucial because the subsequent stages in the process are dependent on the selected concepts. Customer needs are conventionally gathered via survey-based methods, which may require extensive cost to conduct. Along with the massive growth of internet, an alternative to those survey-based methods emerges. Customer needs, as well as other insights about the customers, may be inferred from the opinions, feedbacks, or expectations that customers express in various online channels including online customer reviews. However, the volume and the generating velocity of the online customer review data surpass people's ability to analyze them in a reasonable time. Therefore, in order to utilize online reviews for supporting product designers in decision making, this work proposes methodologies that utilize Natural Language Processing tools, machine learning algorithms, and statistical models. In particular, the methodologies are proposed to support product designers in three specific aspects. First, a methodology is proposed to automatically identify product features that are discussed in the customer reviews as well as their corresponding sentiments. The particular product features that are significantly related to sales rank should become the focus of product designers when considering improvements of the existing product. Second, a novel approach to constructing the choice sets in the absence of both socio-demographic and the actual choice set data is proposed. The choice models that use the proposed choice sets are shown to have better predictive ability than the baseline, i.e., using random choice sets. The choice models with higher predictive ability are useful for product designers to perform demand estimation more accurately. Finally, a methodology is proposed to automatically identify product usage contexts from online customer reviews. Understanding the actual usage contexts is important because it may explain the differences in customer needs, the required design targets, and product preferences. In this work, the identified usage contexts are further complemented by their corresponding aspect sentiments. For product designers, the results enable them to understand customer experience regarding the usage contexts, including the contexts that may not be originally intended by the designers.

*To my family, you are my stars.*

# Acknowledgments

First of all, I would like to thank Professor Harrison Kim for being my academic advisor during my PhD study. I would like to thank you for all the advice, ideas, and discussions regarding the research work that I have done. I am honored to be selected by Professor Kim to join the Enterprise Systems Optimization Lab (ESOL).

I would like to thank all the committee members: Professor Deborah Thurston, Professor Julia Hockenmaier, and Professor Pingfeng Wang. I received valuable questions and suggestions from the committee during the Preliminary Exam that helped me a lot in moving forward with my research that I present in this dissertation. The questions and suggestions during the interesting discussion in the Final Exam were greatly appreciated as well.

I would also thank Fulbright for granting me the chance to pursue a PhD study in the United States, something that I did not even dare to think about before it actually happened. All professors of the Department of Industrial and Enterprise Systems Engineering have been very supportive and I sincerely thank you for allowing me to learn and grow here. The departmental staffs have been helping me a lot, especially Ms. Holly Kizer, Ms. Raymona Wicks, Ms. Barb Bohlen, and Ms. Aleta Lynch.

Over the years, I have been glad to work, befriend, and know the members of ESOL and the people in the TB 416 office, especially Jungmok Ma, Ning Quan, Hyeongmin Han, Jinju Kim, Francois Cluzel, Liang Cong, Michael Saidani, Reza Yousefi Maragheh, and Roshanak Khaleghi. I am thankful for the students who had been working with me for the research, especially Ting-Wai To and Deniz Karaca. I have had great experience in joining the SE 101 instructional team under Dr. James Leake and Dr. Molly Hathaway Goldstein, I would like to thank the fellow TAs for the collaboration and friendship under the Head TA, Hossein Matin.

On a more personal level, I sincerely thank my host family, Jonathan and Sarah DeNeal, and their big family for being very supportive, accepting, and always open-minded. I am thankful for the support from my fellow Fulbright students, especially Richa Niraula, JiHyea Hwang, and Carla Fernandez Corrales; and the fellow Community Aides in Orchard Downs, especially Leonardus and Dorothea Sudibyo, Lisa Chason, and

Tumani Malinga; as well as the friends and neighbors that I have known since I started my life here. Every single of you has become my family here and thank you for being with me during this chapter of my life. Finally, I would like to acknowledge that this dissertation would never materialize without the invaluable support from my family in Indonesia.

# Table of Contents

# List of Tables

# List of Figures

# Chapter 1

# Introduction

The growth of internet has delivered a higher connectivity among people. As the internet technology is getting more accessible, more people are getting connected to one another. Based on the World Bank data on 2016, more than 3 billion people are internet users, which is close to half of the world population. The number has shown a rapid increasing trend since merely two decades ago. Furthermore, the number of internet users has increased not only in particular countries, but it has been observed in all world regions, as seen in Figure 1.1 (Source: [1]).

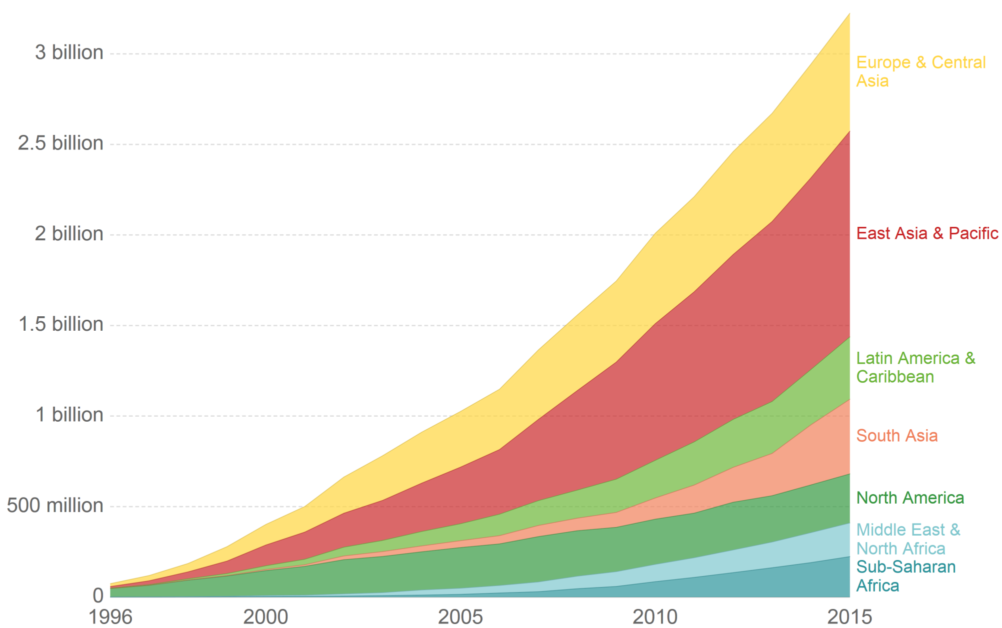



Figure 1.1: The number of internet users by world region

The connectivity opens the opportunity for companies to offer their products and provide their services to customers whom may otherwise be unreachable. The easier access to products and services have attracted more customers to purchase products or services online. The increasing attraction is reflected by the growth of e-commerce sales. As reported by eMarketer [2], e-commerce sales worldwide increased by 24.8% in 2017 compared to the previous year. In terms of total retail sales worldwide in 2017, e-commerce sales claimed

10.2% of it compared to 8.6% in the previous year.

The higher connectivity also allows a more dynamic flow of information. Companies may put information about their products and services on the internet, as well as providing particular customer with particular information that is requested by the customer. On the other hand, customers may also channel their opinions, feedbacks, or expectations towards the products or services offered by companies. For those purposes, customers may use various channels, e.g., customer review section on e-commerce webpages, discussion forums, or blogs; and, by doing so, customers provide valuable information about the products or services. As the number of internet users and e-commerce sales increase, the volume of customer opinions become significant. Due to its massive volume, analyzing voice of customer from online sources requires efficient methodologies that are able to reveal the important insights for the companies.

## 1.1 Product Development Process

In the context of product development process, the voice of customer is an important factor in one of the early phases of the development. According to the generic product development process shown in Figure 1.2 [3], once the target market and business goals have been decided at Phase 0 (Planning), identifying target market's needs becomes the key of Phase 1 (Concept Development). Identifying the needs correctly becomes the foundation of designing product that fulfills the needs, such that the targeted customers are interested to buy the product.

There are three commonly used methods to gather data from customers in order to identify the customer needs, i.e., [3]:

1. Interviews: typically done by discussing the needs with a single customer.
2. Focus groups: typically done by a moderated discussion with a group of 8 to 12 customers. The moderator may be a professional market researcher, or a member of the product development team.
3. Observations: typically done by watching a customer use an existing product to perform particular tasks. The tasks are for which the developed product is intended to be able to perform. Ideally, the observation should be done in the actual use environment. Furthermore, the observation may be either passive or active, in which a member of the product development team works together with the customer to perform the tasks.

While the three methods above are able to gather data from customers directly, the methods may require extensive cost (including time and manpower) to perform. On the other hand, along with the emerging of

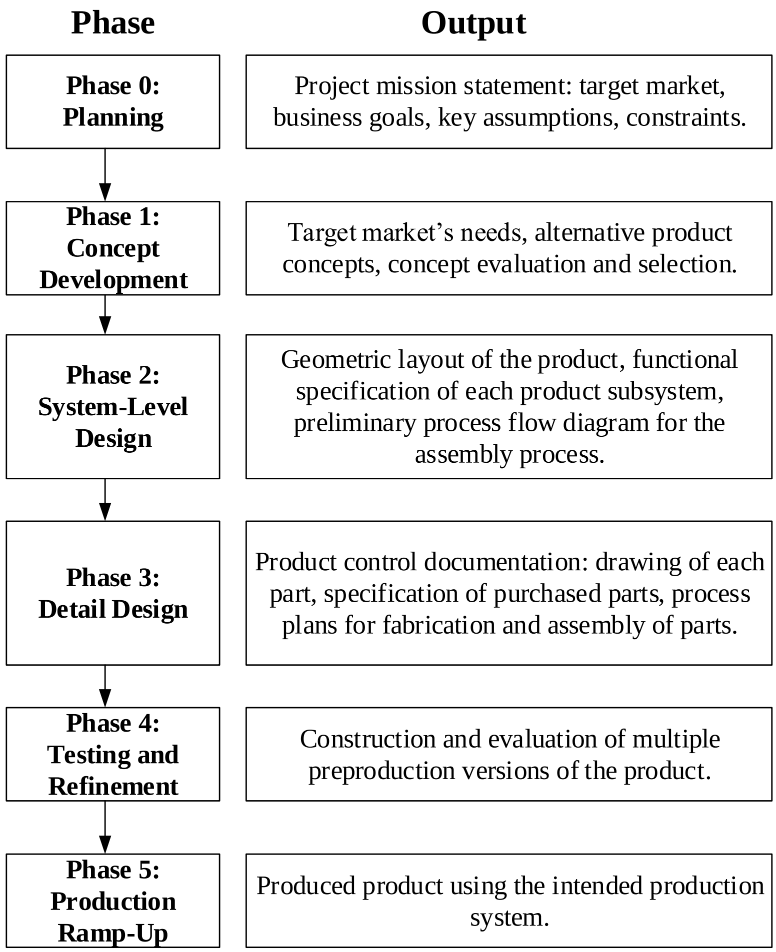



Figure 1.2: Generic product development process

internet technology, customers have been channeling their opinions about products on the internet. Most of the customer opinions are publicly available and the volume is massive. In 2013, in Amazon.com website alone, there had been 35 million customer reviews [4]. Considering the growth of internet users in Figure 1.1 and worldwide e-commerce sales [2], it is reasonable to assume that the volume of customer reviews has been growing bigger as well. Thus, this massive volume of customer reviews may be a valuable source of information to replace or complement the information gathered from the three conventional methods.

## 1.2 Objective and Scope

Considering the opportunities provided by the publicly available data of customer opinions, mostly in the form of customer reviews, this dissertation proposes methodologies to analyze the reviews in order to support decision making in engineering design, which includes the product development process. Regarding its massive volume, the methodologies are designed to perform the analysis in a way that involves as little manual supervision as possible. The results are expected to support decision making in product design process, as either a replacement or complement to the data gathered from the conventional methods, i.e., interviews, focus groups, and observations.

First, a methodology is proposed to show the relation between customer opinions in online reviews and the sales rank. Specifically, customer opinions about particular product features may be related to sales rank, while the opinions about other product features may not. The result should help product designers to focus on the product features that are significantly related to sales rank.

The second methodology is proposed to incorporate customer reviews, which are regarded as a form of online self-presentation, into a choice model. A choice model is frequently used to estimate customer demand with respect to the changes in product attributes. In order to model customer choice more accurately, the differences among customers are captured through the socio-demographic data (e.g., age, gender, income) and incorporated into the model. However, socio-demographic data may be difficult to collect. Moreover, while the purchase data is usually available, the actual choice sets that accompany the purchases are rarely available.

Therefore, in the absence of the socio-demographic data as well as the actual choice set data, the methodology aims to incorporate customer reviews as a representation of differences among customers. The result shows that customer reviews are significantly valuable in differentiating customers and thus constructing choice model with a better predictive ability, compared to not using the information from the reviews. The result should help product designers to build a choice model and estimate the demand with respect to changes in product attributes faster than waiting for the collection of socio-demographic data and actual choice set data through the survey-based methods.

Finally, a methodology is proposed to identify actual product usage contexts from online customer reviews. Based on the identified usage contexts, the aspect sentiment analysis is conducted in order to capture the sentiment towards specific usage contexts. The sentiments may be utilized by product designers to gauge the position of a product relative to its competitors in various usage contexts, as well as verifying customer sentiment towards the intended usage contexts. The result may benefit customers who may filter the products based on the positive sentiment score for the usage contexts that are important for them. The result may also benefit a company or a product designer who may consider to improve the product's performance on specific usage contexts. The company or product designer may as well target a new market segment by designing a product that offers a good performance for a specific usage context in which currently most of the products are perceived negatively by customers. Referring to the product development process in Figure 1.2, the result may become the input for the Phase 1 (possible product improvements) or even Phase 0 (new target market).

The scope of the research in this dissertation would be analyzing textual online customer reviews in free format, i.e. the reviews do not have any structures such as "Pros" and "Cons" sections. The free-format reviews are considered to be more general than the sectioned-format reviews, such that it may be applied to the data from more sources, e.g., Amazon.com, BestBuy.com, etc. However, this research does not pursue the methods to classify the authenticity of the reviews. Hence, all reviews in the data set are assumed to be authentic or the authenticity is assumed by using the reviews that have been verified by the corresponding website as being written by a person who has purchased the product. Furthermore, for the machine learning techniques that require inputs of parameters, e.g., word embedding, this research does not aim to seek the optimal parameter values for those techniques. The research instead focuses on proposing methodologies that require as little supervision as possible and generalizes well into any type of products.

## 1.3 Motivation

The research is motivated by the fact that online review is valuable information that is publicly available in a large volume. The large amount of information produced by customers should be useful to gain insights about customers, which in turn supports decision making in product design. Due to the large size of the data, however, analyzing the reviews and relating them to other data (such as sales rank) are virtually impossible to be performed manually. Furthermore, not only the volume of the data, the generating velocity of online review data surpasses people's ability to analyze them in a reasonable time [5]. Therefore, in order to support product designers in decision making, the proposed methodologies in this dissertation aim to analyze and utilize the data with as little supervision as possible.

The research is also motivated by the challenges to interpret the meaning behind words and sentences in the customer reviews. One of the main challenges is that the free-format reviews may be written in an ungrammatical way. The Natural Language Processing (NLP) tools that are used in this research, such as part-of-speech tagger, dependency tree parser, word embedding; help to identify particular aspects of the reviews. The identified aspects of the reviews help product designers in making design decisions that are supported by large data and may be made faster than waiting for the results from survey-based methods.

## 1.4 Overall Organization

The dissertation is organized as shown in Fig. 1.3 and described as follows. Chapter 1 introduces the background, objective and scope, and motivation of the research. Chapter 2 provides literature review on topics that are related to all remaining chapters, i.e., online review as a form of electronic word-of-mouth, the authenticity issue of online reviews, the recent approaches to product design that utilize online customer reviews, word embedding technique as a Natural Language Processing tool, and X-means clustering method. Chapter 3 presents a methodology to identify the relation between the product features that are discussed in the online customer reviews and sales rank. Chapter 4 proposes a methodology to construct customer choice sets using online data and customer reviews, in the absence of the actual choice sets and socio-demographic data of the customers. The methodology in Chapter 4 applies the methodology in Chapter 3 in order to analyze the product-feature and sentiment words that are discussed in the customer reviews. Chapter 5 proposes the discovery of actual product usage contexts from online customer reviews and identify the corresponding aspect sentiments of the contexts. Finally, Chapter 6 concludes the dissertation and presents future work that may be pursued further.

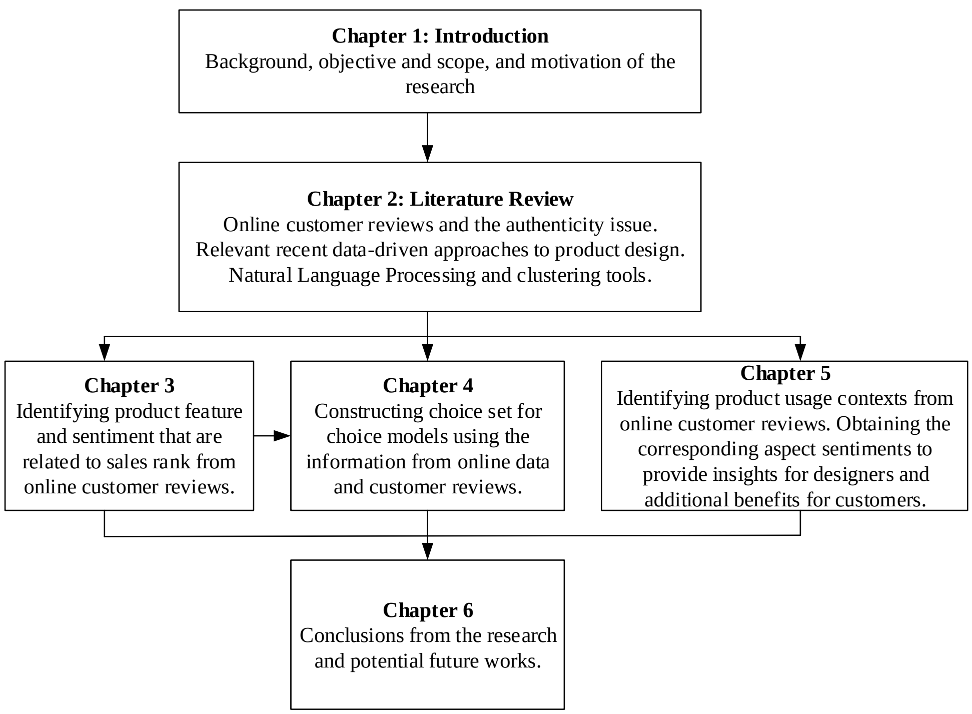



Figure 1.3: Overall organization of the dissertation

## Chapter 2

# Literature Review

## 2.1 Electronic Word-of-Mouth

Word-of-mouth has long been discovered to be one of the significant factors that affect customer buying decisions. In one of the earliest researches about word-of-mouth, the experiment concludes that the exposure to favorable word-of-mouth increases the probability of purchase and vice versa [6]. Among many information channels, word-of-mouth is the one with the highest impact towards customer buying decision, through affecting either awareness or preference [7].

In the internet era, word-of-mouth appears in digitized forms, such as comments, discussions, reviews, and suggestions that are posted online [8]. The digitized word-of-mouth, which is commonly referred to as electronic word-of-mouth, has different characteristics compared to the traditional word-of-mouth, i.e., directed to many individuals, may be anonymous, and may be accessible for a long period of time [9]. Electronic word-of-mouth is defined as the communication about a product made by potential, actual, or former customers, which is made available to a multitude of people via the internet [9]. This becomes a valuable information for customers to make more efficient and rational purchase decisions [10].

Similar to traditional word-of-mouth, electronic word-of-mouth has also become an important factor in shaping customer purchase behavior [11]. It has been reported that 61% of customers consult online reviews before making a purchase [12], 68% of online customers check at least four reviews before making a purchase [13], 80% of customers consult online reviews before making a decision [11], and more than two-third of customers trust online reviews [14]. As reported in a survey conducted by Dimensional Research in 2013 [15], majority of customers acknowledge that reading online reviews impacts their buying decisions. In particular, 90% of customers acknowledge that reading positive reviews impacts buying decisions and 86% of customers acknowledge that regarding to reading negative reviews.

Furthermore, electronic word-of-mouth has become a more important source of information than marketer-generated sources of information, such as advertisements. An experiment discovers that the exposure to online discussions has created more interest towards the products, compared to the exposure to the official information from the company webpages [16]. Another research shows that most respondents feel that electronic word-of-mouth is more important than advertising and they intend to continue doing online discussion with other customers [17].

## 2.2 Online Customer Review Authenticity

### 2.2.1 Definition of Fake Reviews and Difficulties in Detecting Them

Online customer review, which is a form of electronic word-of-mouth, is relatively easy and cheap to write. Consequently, the reviews are written by people with greatly varied expertise and motivation [18]. In particular, there are people whose motivation is not simply sharing their experience but they have an incentive to ensure that some products are reviewed favorably. Those people might then produce inauthentic reviews. Moreover, the production of inauthentic reviews may be supported by the fact that many websites allow reviewers to stay anonymous by displaying the user names only. The motivation behind producing inauthentic reviews may be explained by the discovery that online reviews have raised competitiveness between companies. As the consequence, some companies might produce fake reviews to promote their products and defame their competitors [19].

In the literature, the inauthentic customer reviews have mostly been called as either spam reviews or fake reviews. In Ref. [20], three categories of spam reviews are defined as follows:

1. Untruthful Opinions.

   The reviews that deliberately mislead readers by either promoting an item through undeservedly positive reviews or damaging an item's reputation through maliciously negative reviews. The intention of this category of fake reviews is to influence customer perception towards a product by inflating or damaging the product's reputation [21]. Moreover, these fake reviews are frequently written by reviewers with little or no actual experience with the item being reviewed [22].

2. Reviews on Brands Only.

   The reviews that only comments on the brands, the manufacturers, or the sellers of an item. These reviews are considered as biased, because they do not specifically comment on the item itself.

3. Non-reviews.

   The reviews that contain advertisements or irrelevant contents, such as questions, answers, and random texts.

Among the three categories of fake reviews that are defined in Ref. [20], most of the works in the domain have been dedicated to the first category, i.e., Untruthful Opinions. This category is considered to be the most difficult to identify, either manually or using a machine learning approach. Labeling fake reviews by manual reading is virtually impossible, because a spammer can carefully craft the reviews such that they are similar to the authentic reviews [20]. Most of the fake reviews look perfectly normal until they are compared with the other reviews of the same product [21]. The efforts to conduct the comparisons are non-trivial and it gets even more difficult due to the suspicion that the writers of fake reviews work in groups.

In Ref. [23], a spammer group is defined as a group of reviewers who works together in writing fake reviews. These groups are considered very damaging due to its sheer size, such that they are able to control the sentiment for an item. An example of a set of reviews that is suspected to be produced by a spammer group is shown in Figure 2.1 (Source: Ref. [24]). The figure shows three reviews from each of the three reviewers, i.e., Big John, Cletus, Jake. While each review does not look suspicious –such that it is difficult to manually detecting these reviews as inauthentic; the set of reviews exhibits the following patterns that are commonly found in spammer groups:

1. All group members review the same set of products and give the same rating.
2. All group members post the reviews within a small time window.
3. All group members have the same history of products that they have reviewed so far.
4. All reviews are among the earliest reviews that are posted for the products.

In general, fake reviews are products of a deception process. The accuracy of human deception detection has been found to be only slightly higher than 50 percent, because of the natural truth bias [25]. In an online setting, the detection gets even more difficult due to the absence of the cues to deception in face-to-face communication (e.g., facial expressions, body gestures, tone of voice) and the production of fake reviews that resembles authentic reviews.

### 2.2.2 Approaches to Fake Review Detection

There have been many attempts to detect fake online customer reviews using various approaches. In one of the earliest research in the domain, the approach to fake review detection is based on duplicate detection. In Ref. [20], three types of duplicate and near-duplicate reviews are detected, i.e.,:

1. Duplicate reviews from different users on the same product
2. Duplicate reviews from same user on different products

| | |
|---|---|
| 1 of 1 people found the following review helpful: **Practically FREE music**, December 4, 2004 **This review is from: Audio Xtract (CD-ROM)** I can't believe for $10 (after rebate) I got a program that gets me free unlimited music. I was hoping it did half what was …. | 2 of 2 people found the following review helpful: **Like a tape recorder…**, December 8, 2004 **This review is from: Audio Xtract (CD-ROM)** This software really rocks. I can set the program to record music all day long and just let it go. I come home and my …. |
| 3 of 8 people found the following review helpful: **Yes – it really works**, December 4, 2004 **This review is from: Audio Xtract Pro (CD-ROM)** See my review for Audio Xtract - this PRO is even better. This is the solution I've been looking for. After buying iTunes, …. | 3 of 10 people found the following review helpful: **This is even better than…**, December 8, 2004 **This review is from: Audio Xtract Pro (CD-ROM)** Let me tell you, this has to be one of the coolest products ever on the market. Record 8 internet radio stations at once, …. |
| 5 of 5 people found the following review helpful: **My kids love it**, December 4, 2004 **This review is from: Pond Aquarium 3D Deluxe Edition** This was a bargain at $20 - better than the other ones that have no above water scenes. My kids get a kick out of the …. | 5 of 5 people found the following review helpful: **For the price you…**, December 8, 2004 **This review is from: Pond Aquarium 3D Deluxe Edition** This is one of the coolest screensavers I have ever seen, the fish move realistically, the environments look real, and the …. |

**Big John's Profile** **Cletus' Profile**

| |
|---|
| **Wow, internet music! …**, December 4, 2004 **This review is from: Audio Xtract (CD-ROM)** I looked forever for a way to record internet music. My way took a long time and many steps (frustrtaing). Then I found Audio Xtract. With more than 3,000 songs downloaded in … |
| 2 of 9 people found the following review helpful: **Best music just got …**, December 4, 2004 **This review is from: Audio Xtract Pro (CD-ROM)** The other day I upgraded to this TOP NOTCH product. Everyone who loves music needs to get it from Internet …. |
| 3 of 3 people found the following review helpful: **Cool, looks great…**, December 4, 2004 **This review is from: Pond Aquarium 3D Deluxe Edition** We have this set up on the PC at home and it looks GREAT. The fish and the scenes are really neat. Friends and family …. |

**Jake's Profile**

Figure 2.1: Example of reviews from a suspected spammer group

3. Duplicate reviews from different users on different products

In Ref. [20], the detected duplicate and near-duplicate reviews are labeled as positive examples, i.e., fake reviews. The remaining reviews are all labeled as negative examples. Each review is described by a vector of 35 features that are related to the content of the review (e.g., percent of helpful feedbacks, length of the review title, length of review body, position of the review in the reviews of the product sorted by date, textual features, rating-related features), the reviewer (e.g., ratio of the reviewers' reviews that are among the first reviews of the products, ratio of the reviewers' reviews in which he/she is the only reviewer, average rating given by the reviewer, standard deviation of the ratings), and the product being reviewed (e.g., price, sales rank, average rating). A logistic regression model is subsequently built to learn from the given examples, with the purpose to identify the fake reviews that are not duplicates.

In Ref. [21], four types of behavior are used as the bases to compose the spam score of a reviewer. The four types of behavior are as follows:

1. Writing multiple reviews for the same product
2. Assigning multiple very high or very low ratings to products that share the same attribute (e.g., brand) within a short span of time
3. Assigning ratings that are quite different from the average rating of a product
4. Being a reviewer that writes a review when a product is just available to review and assigning a rating that is quite different from the average rating of the product

For detecting fake reviews posted by groups of spammers, the candidate groups are first identified by using Frequent Pattern Mining [23, 24]. The underlying assumption is that a group of reviewers who frequently posts reviews on the same products are likely to belong to the same group. Similar to the approach to identify a reviewer as a spammer, several types of both group and individual behaviors are proposed as the possible indicators of spamming, i.e.,:

1. Group Spam Behavior
   (a) Time Window: a group of reviewers posts reviews within a short span of time
   (b) Deviation: a group of reviewers assigns ratings that are greatly deviated from the other reviewers
   (c) Content Similarity: a group of reviewers posts reviews that are highly similar
   (d) Member Content Similarity: multiple members of a group of reviewers duplicate or modify their own previous reviews for similar products
   (e) Early Time Frame: a group of reviewers are among the earliest to post reviews for many products
   (f) Size Ratio: the ratio of a group of reviewers to all reviewers of a product
   (g) Size: the size of a group of reviewers
   (h) Support Count: the total number of products of which a group of reviewers has posted reviews
2. Individual Spam Behavior
   (a) Rating Deviation: the deviation of a reviewer's rating compared to the average rating of a product
   (b) Content Similarity: the similarity of reviews posted by a reviewer
   (c) Early Time Frame: a reviewer is among the earliest to post reviews
   (d) Coupling in a Group: a reviewer posts reviews close to the average posting date of his or her group

Both group spam and individual spam indicators are used to calculate three scores of spam contribution, i.e., from a group to a product, from an individual to a product, and an individual to a group. These spam contribution scores are aggregated and subsequently used to rank the candidate groups of spammers. The candidate groups with higher scores are indicated to be groups of spammers.

A different approach using a Bayesian framework is proposed in Ref. [26]. In this framework, the observed author features (e.g., content similarity) and review features (e.g., rating deviation) are assumed to be generated by underlying distributions that take the latent variables as the parameters. It is analogous to the Latent Dirichlet Allocation [27] in topic modeling, in which the observed words are assumed to be generated by underlying distributions that take the latent variables (i.e., topics) as the parameters. In Ref. [26], the latent variables are the "spamicity" of a reviewer and the class of a review, i.e., whether or not a review $r$ is a spam. The main advantage of this work is that it is a fully unsupervised approach, such that it does not require any labeled data.

The literature has stated that detecting fake reviews is a difficult problem. Consequently, it is even more difficult to assess the impact of fake reviews to the outcomes, such as product sales. In Ref. [28], the effect of manipulating reviews is studied. The paper compares reviews from two websites, i.e., TripAdvisor and Expedia. The assumption is that the reviews posted in Expedia are more authentic, because only verified customers are allowed to post reviews. Meanwhile, anybody may post reviews to TripAdvisor. Therefore, the differences in the distributions of the ratings between two websites for the same products are assumed to originate from manipulation, i.e., fake reviews. In the case study, the paper relates the amount of manipulation to the occupancy of the hotels. Based on the case study, the conclusions are:

1. Adding positive fake reviews affects the sales positively up to a certain point. After that point, the effect becomes negative.

2. Deleting negative reviews affects the sales positively up to a certain point. After that point, the effect becomes negative. However, it takes more deletions than additions to reach the turning point because deleting negative reviews is more disguised and thus less likely to be suspected.

3. Excessive addition of positive fake reviews has less negative effect on product sales of strong brands compared to weak brands.

## 2.3 Recent Data-driven Approaches to Product Design

The data-driven approaches that do not rely on collecting data through conventional survey-based methods have been an emerging topic in the design domain. These approaches utilize publicly available online customer reviews as well as the non-textual parts of the reviews, such as the overall rating, the helpfulness rating, etc., in order to support the product development process. In a review paper on recent advances in mining big consumer opinion data for product design, a framework is proposed to categorize various topics of research that have been conducted in this domain, as shown in Figure 2.2 (Source: [5]). As shown on the top part of Figure 2.2, there are two main activities, i.e., mining and processing the consumer opinion data and utilizing the data for product design. A number of works are presented and discusses in this subsection as the representatives of the relevant researches.

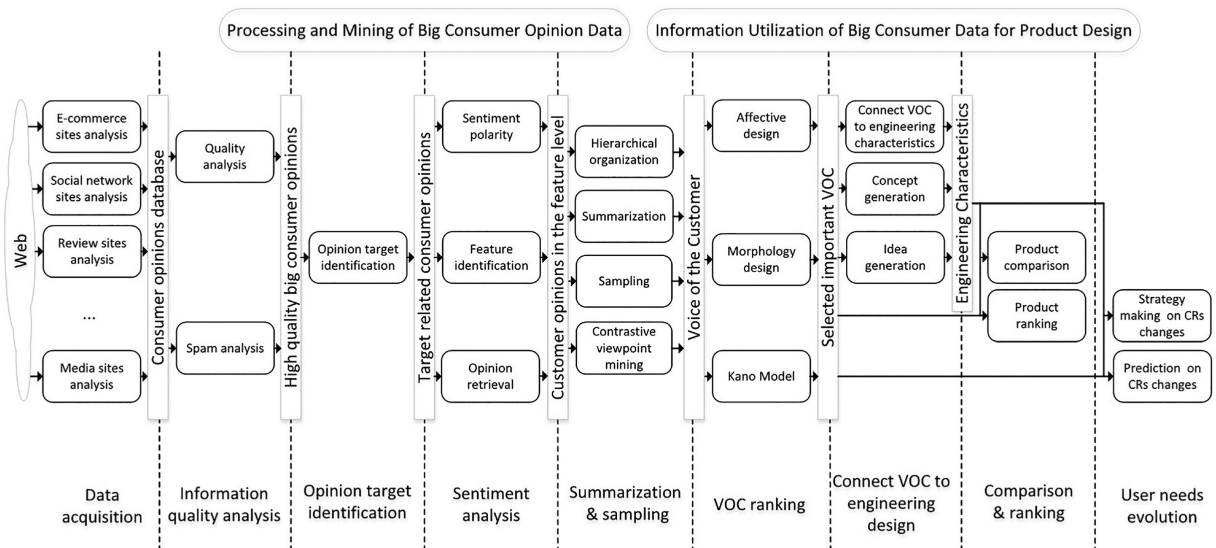



Figure 2.2: A framework of mining online customer data for product design

A number of recent researches propose methodologies to acquire the data efficiently, describing the relations between the entities in the data, and filter the data. In the research related to data acquisition, Lim and Tucker [29] develop a Bayesian-sampling-based methodology to identify the optimal search keyword combinations that maximize the returned product-feature-related data. They remark that the quality of the identified search keyword combinations relies heavily on the first search keyword, such that selecting the first becomes an essential problem to be solved in the future. In the research related to describing the entities in the data, Shi et al. [30] propose a text mining methodology that utilizes part-of-speech tags and collocations to build a network that relates the knowledge concepts in design and engineering, based on the acquired data. The nodes in their network are the concepts, which are obtained from the subject and object of a sentence, and the edges between nodes are constructed whenever two concepts appear in the same sentence.

In the research related to data filtration, which in Figure 2.2 is included to the stage of Information Quality Analysis, Zhang and Tran [31] propose a helpfulness score to filter online customer reviews. The score is calculated based on the information gain of the words in a review and the number of votes that a review receives for being helpful. Zheng et al. [32] propose a semi-supervised method to classify online customer reviews into high quality (useful) and low quality (spam or containing little information). Qi et al. [33] filter online customer reviews by predicting their helpfulness using five categories of features, including linguistic features. Based on the filtered reviews, product attributes are identified and they are weighted by their sentiments. Finally, the product attributes are mapped into Kano's model in order to classify the attributes into categories such as attractive, one-dimensional, and must-have.

The spam analysis at the Information Quality Analysis has been discussed at Section 2.2. The activities in the Opinion Target Identification and Sentiment Analysis stages are performed in the proposed methodologies in the following chapters.

In the research that apply the data for product design purposes, online customer reviews are used to measure the attractiveness of new product function candidates, relate product features and sales rank, determining preliminary design specifications, and evaluate design alternatives. Zhang et al. [34] predict the attractiveness of new product function candidates for a particular user by predicting the user's rating towards the new function. Their methodology uses both online customer reviews (for calculating product similarity) and survey data (for calculating rating candidate functions and user similarity). The main disadvantage of the method is that it only provides the prediction for one particular user, yet it requires surveying a number of users that are considered similar to the targeted user.

Suryadi and Kim [35] propose a methodology to identify product features that are significantly related to sales rank. The product features are automatically identified from the customer reviews using Natural Language Processing tools, such as part-of-speech tags and word embedding. The occurrences of the product-feature words in the reviews are related to the sales rank using a linear regression model.

Chaklader and Parkinson [36] propose a metric called Weighted Phrase Rating that is used to determine preliminary design specifications. The metric uses the average rating of the reviews that contain specific cue phrase, e.g., "tight", and is compared to the overall average rating of the product. Based on the comparison and the specifications of the products, product designers may infer the preliminary acceptable design specifications.

Chiu and Lin [37] evaluate design alternatives using a regression model that is constructed based on online customer reviews. In their regression model, the specifications of the design alternatives become the independent variables. The sentiments of the selected adjectives from the online customer reviews become the dependent variables. The subjectively selected adjectives and other subjective inputs or decisions become the main disadvantages of their methodology.

## 2.4 Natural Language Processing

Most of online customer review data are in the forms of texts. In order to analyze the data, the texts are parsed into sentences, then the sentences may be parsed further into phrases or words. One of the basic ways to classify words is using part-of-speech. Parts-of-speech are classes of words that have similar function with respect to the words that occur nearby or to the affixes they take [38]. The parts-of-speech may be utilized for various purposes. For example, in this research, the parts-of-speech that are used to analyze product reviews are noun and adjective. Nouns are used to identify product-feature words and adjectives are used to identify sentiment words.

When the words form a sentence, a dependency tree can describe the structure of it by relating words in terms of binary semantic or syntactic relations [38]. Therefore, each link in the tree explains the relation between two words. The advantage of using the tree over the bag-of-words approach, i.e. treating a sentence as a linear sequence of words, is its ability to describe relations between words, regardless of the distance between the words, as shown in Reference [39]. Therefore, as in Reference [40] and Reference [35], this research uses dependency tree to identify the related words in a sentence, e.g., product-feature and sentiment words in a review sentence.

### 2.4.1 Word Embedding

Word embedding is a distributed representation for words in a vector space [41]. It is based on the idea that similar words have similar distribution of words that are likely to appear along with them. Therefore, the vectors representing similar words should be similar as well. Zhang et al. [42] has applied word2vec, a word embedding tool, to retrieve synonyms of a given set of product-feature words. *In this research, however, the product-feature words are initially unknown. The word embedding tool is then applied to discovering those words from customer reviews.*

Two models to learn good word embedding are introduced by Mikolov et al. [41], i.e. Continuous bag-of-word and Skip-gram models. In Continuous bag-of-word model, the objective is to maximize the probability of observing a target word given context words. On the other hand, Skip-gram model's objective is to maximize the probability of observing context words given a target word. Context words are commonly defined as the words surrounding a target word within a window of words. Context words may be defined differently, such as considering the dependency relations between words in a sentence [39], but this research uses the window-based definition.

The word embedding is learned by training a neural network. As illustrated in Figure 2.3 for the Continuous bag-of-word model, a network consists of three layers, i.e. Input, Hidden, and Output layers.

The figure is modified from Ref. [43] to clarify the notations. In Input layer, each of the C context words is represented by a one-hot V-dimensional vector; where V is the vocabulary size of the corpus. A one-hot vector means that the value for all dimensions equals 0, except for the dimension corresponding to the context word that equals 1. For example, if the first context word is the k-th word in the vocabulary, then $x_1 = [x_{11}, x_{12}, ..., x_{1k}, ..., x_{1V}]^T = [0, 0, ..., 1, ..., 0]^T$

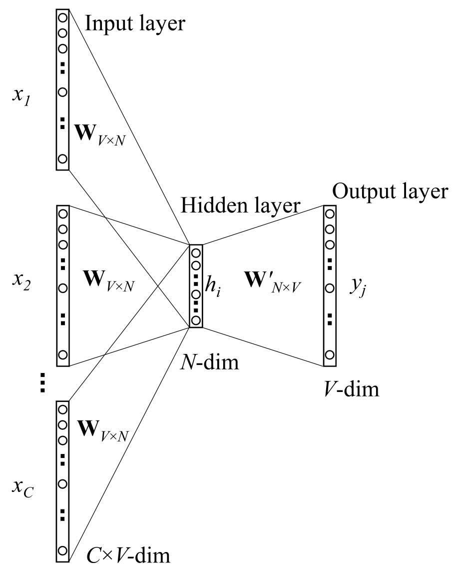



Figure 2.3: Continuous bag-of-word model

From Input layer to Hidden layer, an x vector is transformed by a (V x N) input matrix $W$, as shown in Equation 2.1. Each row in $W$ represents the embedding of a word. For word $I$ that is represented by a row vector $v_{w_I}$ in W, the transformation results in an $N$-dimensional $v_{w_I}^T$ vector in Hidden layer. In the context of multiple context words, the Hidden layer takes the average of the results, as shown in Equation 2.2.

$$h_I = W^T x_I \tag{2.1}$$

where:

$h_I$ = the hidden layer vector corresponds to an input word $I$

$x_I$ = one-hot (V x 1) vector that represents word $I$

$W$ = the (V x N) input matrix

$$h = \frac{1}{C} W^T (x_1 + x_2 + ... + x_C) = \frac{1}{C} (v_{w_1} + v_{w_2} + ... + v_{w_C})^T \tag{2.2}$$

where:

$C$ = number of context words

$v_{w_c}$ = (1 x N) vector that represents word $c$ in the input matrix W, c = 1, 2, ..., C

$x_I$ = one-hot (V x 1) vector that represent word $I$

$W$ = the (V x N) input matrix

From Hidden layer, the vector is further transformed by an (N x V) output matrix $W'$ into the Output layer, as shown in Equation 2.3. Each column in $W$ corresponds to a word in the vocabulary. For word j that is represented by a column vector $v'_{w_j}$ in W', the transformation results in a scalar $u_j$ as shown in Equation 2.4, which does not have any significant meaning by itself. It becomes meaningful once it is transformed using softmax function in Equation 2.5 into $y_j$ which denotes the probability of word j being the target word given the context words in Input layer.

$$u = W'^T h \tag{2.3}$$

where:

$h$ = the hidden layer vector

$W'$ = the (N x V) output matrix

$u$ = the (V x 1) output vector

$$u_j = v'^T_{w_j} h \tag{2.4}$$

where:

$h$ = the hidden layer vector

$v'_{w_j}$ = the j-th column of output matrix W' that corresponds to word $j$

$u_j$ = the j-th element in the output vector u

$$P(w_j|w_1, w_2, ..., w_C) = y_j = \frac{exp(u_j)}{\sum_{j'=1}^{V} exp(u_{j'})} \tag{2.5}$$

where:

$C$ = cardinality of context words

$u_j$ = the value of $j$-th element of output vector

$w_j$ = word $j$ in the vocabulary

$y_{i,j}$ = probability of word $j$ being a context word at the $i$-th output vector

In order to learn high-quality vector representations of words [44], the objective of the learning model is quantified as maximizing the average log-likelihood of a sequence of training words $w_1, w_2, \ldots, w_T$ [41, 43], as shown in Equation 2.6. For each $w_t$ in the training set, its context words are known based on its surrounding words within a window, and thus the model learns via neural networks to maximize the probability assigned to $w_t$ given its context words.

$$\frac{1}{T}\sum_{t=1}^{T} E_t = -\frac{1}{T}\sum_{t=1}^{T} \log P(w_t|w_1, w_2, \ldots, w_C) \tag{2.6}$$

where:

$C$ = cardinality of context words

$E_t$ = loss function for word $t$ in the training set, t = 1, 2, ..., T

$w_t$ = the $t$-th word in a sequence of words

The optimization the model based on maximizing Equation 2.6 may be done by performing gradient descent. To improve the input matrix W, which records the word embedding for each word in the vocabulary, a backpropagation procedure is performed. There are methods that can be used to speed up the optimization, such as Hierarchical Softmax and Negative Sampling [43].

## 2.5 X-means Clustering Method

Clustering refers to dividing the data into groups, such that each group, or cluster, contains items that are similar to one another and dissimilar to items in other groups. It may also be viewed as an unsupervised learning of a hidden data concept. Clustering techniques are traditionally categorized into two types, i.e., hierarchical and partitioning [45]. Hierarchical clustering may start with each cluster containing a single data and recursively merge two or more of the most similar clusters. It may also start with all data points in a single cluster and recursively split the cluster. On the other hand, partitioning clustering techniques iteratively reassign data points between the clusters in order to gradually improve the clustering quality. The X-means clustering technique that is used in this research belongs to the partitioning category.

X-means clustering is proposed to overcome the disadvantages of K-means clustering, which is a partitioning clustering and by far the most popular clustering tool in scientific and industrial applications [45].

The disadvantages of K-means are [46]: (1) it is slow, (2) it requires the number of clusters K to be provided by the user, and (3) it obtains worse solution using a fixed value of K compared to being able to change K dynamically.

X-means clustering uses K-means clustering as its basis. The algorithm starts with K equals to the lower bound of the specified range of K, then the following steps are performed iteratively [46]:

1. Performing K-means clustering to obtain the cluster center and assignment.
2. Performing temporary cluster center splitting and assessing if a split results in better performance metric, i.e. Bayesian Information Criterion (BIC). This step is illustrated in Figure 2.4. If it does, a split is made permanent.
3. If the number of cluster centers has exceeded the upper bound of the specified range of K, the algorithm stops. Otherwise, it goes back to Step 1.

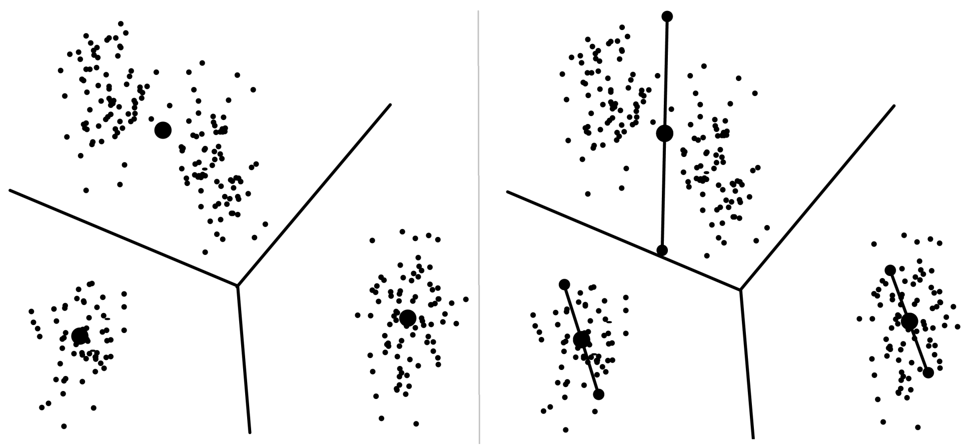


Figure 2.4: An example of cluster center splitting in X-means clustering

The formula of Bayesian Information Criterion (BIC) in Equation 2.7 is used as a measure to assess whether or not cluster centers should be split. In the formula, a model refers to a particular clustering structure.

$$BIC(M_j) = \hat{l}_j(D_{BIC}) - \frac{p_j}{2}\log(R) \tag{2.7}$$

where:

$BIC(M_j)$ = the Bayesian Information Criterion value of the $j$-th model in X-means clustering

$\hat{l}_j(D_{BIC})$ = log likelihood of data $D$ in BIC computation

$p_j$ = number of parameters in the $j$-th model in X-means clustering

$R$ = number of data points

The log likelihood of a data set is computed using Equation 2.8. Under the identical spherical Gaussian assumption, the probability of each point in the data set is defined as $P(x_i)$ in Equation 2.8 and the variance of a data set is formulated in Equation 2.9. Substituting the estimated variance from Equation 2.9 to Equation 2.8 and simplyfing the equation, the log likelihood of data points in cluster n is shown in Equation 2.10.

$$l(D) = \log \Pi_i P(x_i) = \sum_i \left( \log \frac{1}{\sqrt{2\pi}\sigma^M} - \frac{1}{2\sigma^2} \|x_i - \mu_i\|^2 + \log \frac{R_{(i)}}{R} \right) \tag{2.8}$$

where:

$l(D)$ = log likelihood of the data set

$M$ = the number of dimensions of a data point

$\mu_i$ = the center of the cluster in which data point i belongs

$P_{x_i}$ = the probability of data point $x_i$ with respect to the cluster center $\mu_i$

$R$ = the cardinality of the data set

$R_{(i)}$ = the cardinality of data point i, i.e. 1

$\sigma$ = standard deviation of the data set

$x_i$ = data point i

$$\hat{\sigma^2} = \frac{1}{R-K} \sum_i \left( x_i - \mu_{(i)} \right)^2 - \frac{R_n - K}{2} + R_n \log(R_n) - R_n \log R \tag{2.9}$$

where:

$K$ = the number of clusters in the data set

$\mu_i$ = the center of the cluster in which data point i belongs

$R$ = the cardinality of the data set

$R_n$ = the cardinality of the cluster n

$\hat{\sigma}^2$ = estimated variance of the data set

$x_i$ = data point i

$$\hat{l}(D_n) = -\frac{R_n}{2} \log(2\pi) - \frac{R_n.M}{2} \log(\hat{\sigma}^2) - \frac{R_n - K}{2} + R_n \log R_n - R_n \log R \tag{2.10}$$

where:

$\hat{l}(D_n)$ = log likelihood of the cluster n

$K$ = the number of clusters in the data set

$M$ = the number of dimensions of a data point

$R$ = the cardinality of the data set

$R_n$ = the cardinality of the cluster n

The selection of X-means clustering technique in this research has also considered the diadvantages of applying hierarchical clustering [45]. First, selecting the appropriate stopping criteria in hierarchical clustering is difficult. Second, hierarchical clustering often requires a connectivity matrix, i.e., a matrix of distances or similarities between all data points in the training set. As the result, for a large data set, the matrix may be very large and thus becomes impractical to store and utilize.

Chapter 3

# A Methodology for Identifying the Relation between Online Customer Reviews and Sales Rank

## 3.1 Introduction

Products are designed and manufactured to be successful in the market, i.e. customers are willing to buy the products. However, the buying decision process is complicated to observe and model explicitly. The underlying psychological processes, such as motivation, perception, learning, and memory [47], which affect a buying decision, differ by individuals and situations. To describe a general buying decision process, a five-stage model has been proposed [47]. The model is represented in a diagram shown in Figure 3.1.

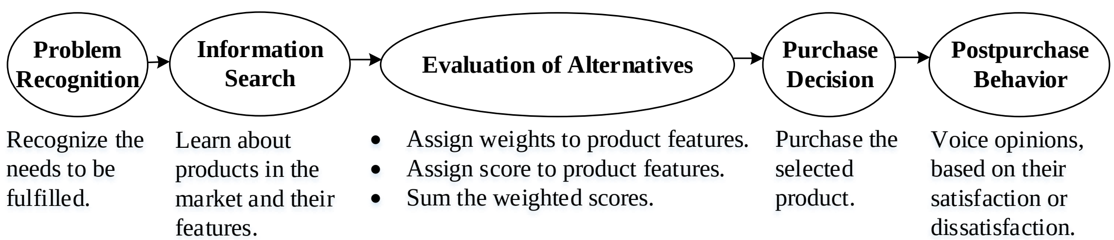



Figure 3.1: Five-stage model of buying decision process

Based on the five-stage model shown by the diagram in Figure 3.1, customers start the process by recognizing the needs (Problem Recognition stage) and followed by searching for information about alternatives that may fulfill those needs (Information Search stage). Therefore, in order to trigger a purchase decision, it is essential for product designers to firstly identify the needs of customers in the target market. Furthermore, designers need to obtain the weights that customers assign to product features (Evaluation of Alternatives stage), which reveal the importance of each product feature for the customers. Finally, designers need to collect the feedback from customers who have either purchased or had experience with the product or a similar one (Postpurchase Behavior stage), in order to understand the possible ways to improve the product. For those purposes, product designers may conduct interviews, surveys, focus group discussions, etc. These methods, however, can be time-consuming, labor-intensive, and expensive to conduct [48].

As an alternative to the aforementioned conventional methods, analyzing publicly available online customer reviews is a resource-efficient method to learn customer needs and preference. Online reviews have grown to become an important source for customers to do information searches about product quality, substituting and complementing other forms of communication, e.g., business-to-consumer [49]. As reported in Ref. [13], 68% of online customers check at least four reviews and almost 25% of them check at least eight reviews before buying. Although there has been a stream of research dedicated to verify the authenticity of product reviews, as initialized by [50], this chapter limits the scope of the research by assuming that the reviews are written voluntarily and thus can be considered authentic [48].

In the framework of the five-stage model and an e-commerce setting, online review may be one of the inputs for the Evaluation of Alternatives stage. The processes in the evaluation stage are hidden, but the input and the resulting purchase decision are both observable. The proposed methodology in this chapter aims to systematically reveal one of the processes at the evaluation stage, i.e. assigning weights to product features. By discovering product features that are significantly related to product sales, it may be implied that those features are the ones weighted as more valuable by customers. Thus, this information provides an objective data-driven suggestion for designers about possible features to improve.

In revealing the hidden process of assigning weights to product features at the evaluation stage, there are challenges as follows:

1. Customers can discuss product features that are not mentioned in the product description on a product's webpage. On the other hand, customers may not discuss product features that are mentioned in the product description. Therefore, product descriptions are not adequate to capture the product features discussed in the reviews.

2. Customers can discuss the same product feature using different words, e.g., "drive", "storage", and "SSD" refer to the same product feature in a laptop.

3. Customers can express their opinions with their own words and expressions in free-format reviews. Thus, free-format reviews are obviously more difficult to analyze, compared with reviews that have been distinctly divided into Pros and Cons sections, such as in Ref. [51].

Regarding the challenges above, the proposed methodology needs to solve four tasks, i.e.:

1. The methodology has to obtain product-feature words (*Task 1*), i.e. words that represent product features, that are actually discussed in the customer reviews.

2. The methodology has to group the same product-feature words that refer to the same product feature (*Task 2*).

3. The methodology has to obtain words that describe sentiment, as well as the intensity of the sentiment (*Task 3*), due to the fact that the free-format reviews are not explicitly divided into Pros and Cons sections.

4. The methodology has to connect each sentiment word in a sentence to the corresponding product-feature word in the sentence (*Task 4*), due to the fact that the dependency between words in a sentence may not be as straightforward as, for example, the adjacency between those words.

There has been research done on the similar topics, as discussed in more detailed in Section 3.2. However, there are key differences between the proposed methodology in this chapter and the previous works, i.e.: (1) the application of word-embedding followed by X-means clustering to automatically obtain product-feature words, (2) the analysis of free-format review data that are not divided into Pros and Cons sections, (3) the elaboration of methods applied in each stage of the methodology, and (4) the analysis of review title as a separate variable from the review content to discover its importance compared to the content. Furthermore, subjective inputs, judgments, and decisions are kept to minimum in the proposed methodology. It does not require, for example, words as initial seeds to discover product-feature words or human judgments (e.g., crowdsourcing). Therefore, the methodology is replicable and generalizable to data sets of different products. It is an improvement to the methodology used in the initial research [52, 53] that mainly relies on subjective judgments in identifying and grouping the relevant product-feature words.

This chapter is organized as follows. Section 3.2 elaborates the previous works in the similar topic, followed by the introduction of word embedding technique. Section 3.3 details each stage in the proposed methodology. Section 3.4 describes the data used for case study, shows the results from processing the data, and finally presents the regression results. Section 3.5 discusses both the methodology and the results. The last section concludes the chapter.

## 3.2 Literature Review: Previous Works in Solving the Four Tasks

This section is focused on the four tasks, which have been presented in the previous section, that are required to interpret the textual content of free-format reviews. It is worth mentioning here that there are papers that completely ignore textual contents of online customer reviews and only utilize variables such as number of reviews and star ratings [49, 54]. Interestingly, one of the results in Ref. [49] suggests that customers actually read the review content. *Consequently, this chapter argues that the inclusion of textual-related variables in* *the analysis is necessary.*

In this research, interpreting free-format reviews starts with obtaining product-feature words. In some of the previous researches, the product-feature words have been known or predetermined, such as in Ref. [55]. When they are not known, as the case in this chapter, various approaches have been applied, as summarized in Table 3.1. Many of those approaches rely upon manually annotated data, as well as subjective predetermination of linguistic patterns and product-feature words to obtain. The main disadvantage of the heavily manual approaches is that the methodology may not be generalizable into the data from other domains, e.g., different product reviews.

Table 3.1: Summary of previous approaches to solve Task 1

| References | Approaches | Disadvantages |
|---|---|---|
| [56] | Subjective determination | Depending highly on the person who annotates the corpus |
| [57, 58] | Supervised machine learning tools: Decision Stump, Conditional Random Fields | Requiring manually annotated or tagged training data |
| [59, 60, 61, 62] | Association rule to find noun or noun phrases that frequently appear together | Resulting in nouns with high frequency, but not related to product features [63] |
| [63] | Association rule with additional filtering step using a set of "subjective adjectives" | Requiring manually constructed set of "subjective adjectives", which can be domain-specific for a particular type of products |
| [64] | High tf.idf (term frequency, inverse document frequency) rule | Resulting in nouns with high tf.idf, but not related to product features |
| [65, 40] | Part-of-speech (POS) patterns | Requiring manually determined POS patterns to mine |
| [66] | Hidden Markov Model, based on tags of product-feature and sentiment words | Requiring manual tagging of training data |
| [67] | Latent Dirichlet Allocation (LDA) | Requiring predetermined number of topics to generate (in this approach, a topic corresponds to a product feature) |
| [68] | Augmented LDA to learn both product-feature and sentiment words | Requiring predetermined number of topics to generate |

The approaches to solve Task 1 that rely less on manual involvement apply association rule, tf.idf (term frequency, inverse document frequency), and LDA (Latent Dirichlet Allocation). Association rule is used to find frequent *itemsets*, i.e. words or phrase that occurs together frequently [59, 60]. The frequent *itemsets* are assumed to be product-feature words. In fact, as explained in Ref. [63], that might not be the case and the proposed pruning rules are not able to filter the irrelevant *itemsets*. The same disadvantage applies to the approach that assumes words with high tf.idf to be product-feature words [64]. For the LDA-based approaches [67, 68], the main disadvantage is the necessity to determine the number of topics beforehand.

In the case of online reviews, the number is not known beforehand, because customers might discuss product features that are not described in the product's webpage. *Considering the disadvantages of the previous approaches, the proposed methodology aims to obtain product-feature words with as little manual involvement as possible and exploit the review data to guide the process.*

Performing Task 1 often returns an unmanageable number of product-feature words. However, in fact, many of those words refer to the same product feature, e.g., "screen" and "monitor" in a laptop. Therefore, for the purpose of interpreting the reviews as accurately as possible, it is essential to solve Task 2. Table 3.2 summarizes the previous approaches to solve Task 2.

Table 3.2: Summary of previous approaches to solve Task 2

| References | Approaches | Disadvantages |
|---|---|---|
| [56] | Subjective grouping | Depending highly on the person who groups the words |
| [40] | Product ontology | Requiring manually constructed ontology |
| [51] | Multilevel LDA | Requiring predetermined number of topics to generate |
| [62] | WordNet-based similarity | Requiring word sense disambiguation to use WordNet in order to determine the correct similarity between words |
| [61] | WordNet-based similarity and agglomerative clustering | Requiring word sense disambiguation to use WordNet and the details for clustering are not provided; |
| [69] | Lexical similarity and Expectation Maximization | Assuming good quality product-feature words have been obtained and the number of groups is known beforehand |

It can be seen from Table 3.2 that many approaches require subjective decisions, such as determining the number of product-feature groups. In reality, the number of product features that are discussed in the product reviews is initially unknown. Therefore, a clustering tool such as K-means clustering is not suitable, because the number of clusters $K$ needs to be determined. On the other hand, X-means clustering does not require the number of clusters as an input. Iteratively, X-means clustering splits a cluster temporarily into two and computes the BIC (Bayesian Information Criterion) measure in Equation (2.7). A cluster is permanently split only if there is an improvement from splitting, i.e. the BIC value increases. The iteration stops when there is not a split of clusters that increases the BIC value. Therefore, the clusters can be finally obtained without predetermining the number of clusters.

One of the objective approaches in Table 3.2 is using WordNet-based similarity. However, a similarity-based approach in WordNet requires word sense disambiguation [38] technique to obtain the correct similarity between a pair of words. For example, the similarity between the words "battery" and "computer" in WordNet depends on the sense of both words. If "battery" is defined in the sense of "a device that produces electricity" and "computer" is "a machine for performing calculations automatically", then the similarity is significantly higher than if "battery" is defined in the sense of "an assault in which the assailant makes physical contact". The common practices of averaging or taking the maximum similarity values might be misleading in reflecting the similarity between two words. In this research, word embedding is utilized in order to capture the similarity between words in the customer reviews. *Therefore, in order to overcome those aforementioned disadvantages, the proposed methodology combines a word embedding and X-means clustering approaches in order to solve Task 1 and Task 2 automatically and objectively, i.e., without manually annotating training data, predetermining linguistic patterns, or predetermining the number of product-feature words.*

While Task 1 and Task 2 deal with product-feature words, Task 3 deals with sentiment words. Table 3.3 summarizes the previous approaches to solve Task 3. Some approaches rely on manually annotated data [70, 57], or subjective and domain-specific inputs (e.g., initial seeds of patterns for pattern-based search [55]). Other approaches assume that the adjective closest to a product-feature word is the one explaining the product-feature word [59, 60]. However, that distance-based assumption might not be the case, and hence dependency tree is used [40]. An approach assumes that the most frequent adjectives are considered as sentiment words [56]. Nevertheless, the assumption is arguable, because the frequency of an adjective does not necessarily reflect its sentiment intensity. Finally, the approach that uses WordNet [64] has a disadvantage that has been previously explained. *In order to overcome those disadvantages, the proposed methodology simply identifies the adjectives as sentiment words. The identification of adjectives is objectively obtained from a part-of-speech tagger.*

Table 3.3: Summary of previous approaches to solve Task 3

| References | Approaches | Disadvantages |
|---|---|---|
| [70, 57] | Manual annotation of the sentiment words and polarity | Depending highly on the person who annotates the corpus |
| [61] | Amazon Mechanical Turk | Depending highly on the people who join the crowdsourcing |
| [59, 60] | The adjective closest to a product-feature word is considered as a sentiment word | Assuming an adjective always modifies the closest noun; not using sentiment intensity quantification |
| [56] | The most frequent adjectives are collected, and considered as sentiment words | Requiring a subjective threshold for the frequency of adjectives; not using sentiment intensity quantification |
| [55] | Pattern-based search | Requiring initial patterns, e.g., "the (feature) is (sentiment)" |
| [64] | Senti-WordNet (the complement of WordNet, with added sentiment polarity) | Requiring word sense disambiguation to use WordNet in order to determine the correct sense for the word |
| [40] | Dependency tree | Not using sentiment intensity quantification |

Furthermore, in order to capture a customer's sentiment more accurately, it is important to quantify the sentiment intensity. For example, a comment of "*great* battery" is more intense than "*good* battery". *In the proposed methodology, the sentiment intensity quantification is obtained from SenticNet4 dictionary, which captures the denotative and connotative information associated with objects, people, actions, and events* [71].

Finally, after product-feature words are grouped (Task 2) and sentiment words are identified (Task 3), the correct connection between those words needs to be identified (Task 4). In the previous works, as summarized in Table 3.4, other than manual mapping of the connection between sentiment word and its corresponding product-feature word [70], either distance or dependency is used to infer the connection. In the distance-based approach, a sentiment word is simply connected to the closest product-feature word [59, 60]. In the dependency-based approach, several rules are applied to a dependency tree in order to obtain connected product-feature and sentiment words [56, 57, 40, 61]. Regardless of the distance between words in a sentence, a dependency tree is capable to show the words that are related, as discussed later in Section 3.3.4 and presented in Figure 3.5. *Considering its advantage compared to the distance-based approach, the proposed methodology uses a dependency tree to infer the connections between product-feature and sentiment words.*

Table 3.4: Summary of previous approaches to solve Task 4

| References | Approaches | Disadvantages |
|---|---|---|
| [70] | Manual mapping of sentiment words to its corresponding product-feature words | Depending highly on the person who annotates the corpus |
| [59, 60] | Distance-based approach | Assuming the sentiment word is related to the closest product-feature word |
| [56, 57, 40, 61] | Dependency-based approach | Requiring rules to identify the related sentiment and product-feature words from a dependency tree |

## 3.3 Methodology

The proposed methodology is presented as a flowchart in Figure 3.2. The parenthesized numbers on the flowchart correspond to the corresponding subsections of this section.

### 3.3.1 Data Preprocessing

The main inputs of this methodology are online customer review, sales rank, and price data. The additional inputs are product manual documents, or similar documents that objectively describe a product, and a sentiment dictionary, such as SenticNet4. Of all inputs, customer review data is the one that requires the

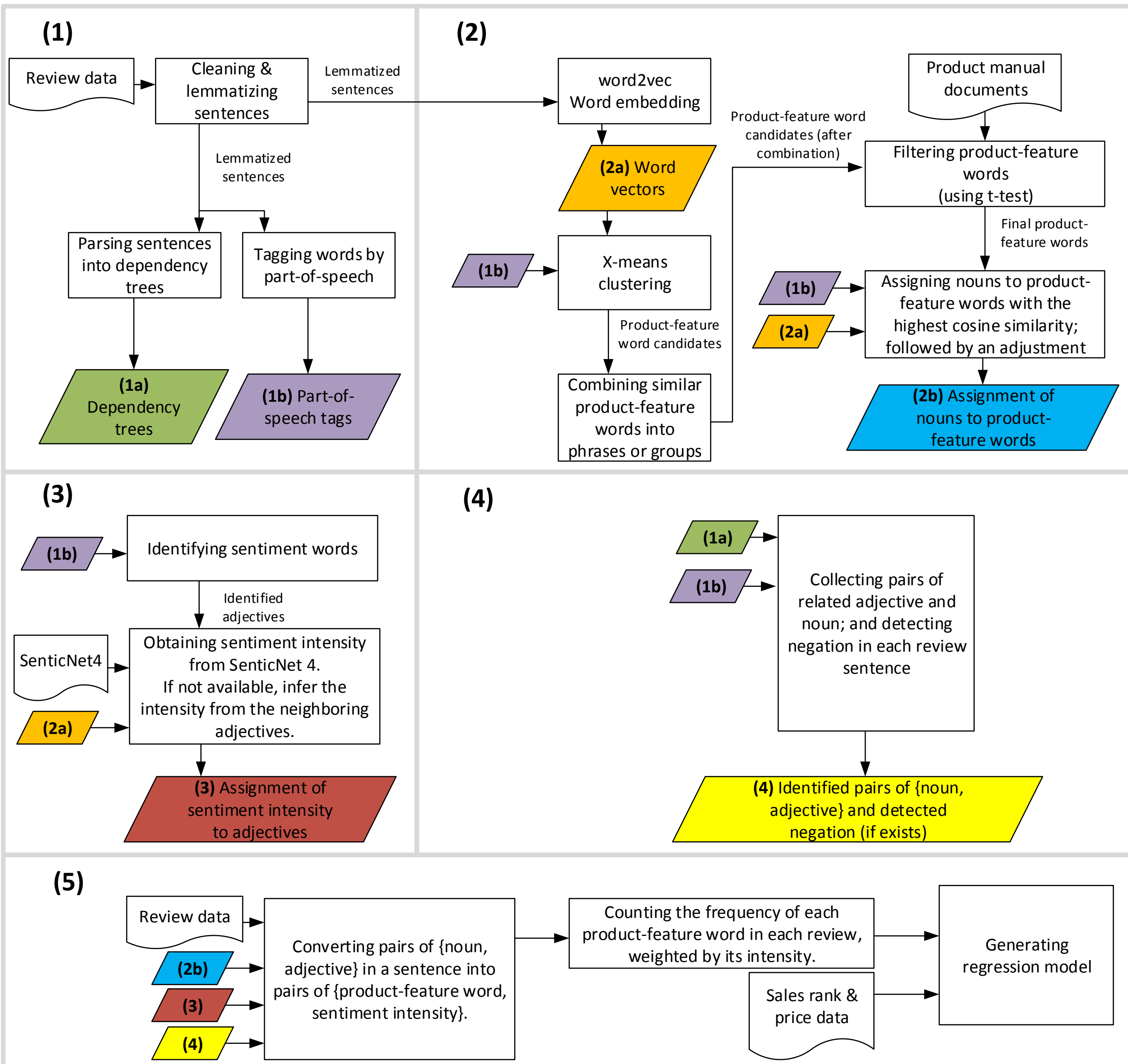



Figure 3.2: Methodology for identifying the relation between online customer reviews and sales rank

most preprocessing, as described in this subsection. The preprocessing for other inputs is relatively trivial, e.g., removing commas and dollar signs from the price data, such that “$1,200” may be recognized as a number by the software.

The first step in preprocessing the customer review data is removing non-alphanumeric characters, such as #, $, and %. These characters are considered not helpful to reveal either product-feature or sentiment words from a sentence. Afterwards, a lemmatizer, which is obtained from NLTK (Natural Language Toolkit) package in Python [72], is applied to replace various word forms into their basic forms, e.g., replacing a word in plural form ”years” into ”year”. The replacement is required to avoid having the same word in different forms embedded into different vectors in the later stage of the methodology. Since the lemmatization is applied before part-of-speech tagging, the NLTK lemmatizer in Python would mostly replace the words in plural forms with singular forms. Without the information of the part-of-speech tags, it does not replace the words with their lemmas, such as “playing” with “play”, or “better” with “good”. Therefore, since this step only replaces the words in plural forms, the dependency tree parser and part-of-speech tagger in the next step are expected to perform well, i.e., successfully identifying the correct relation between words and the part-of-speech tags of the words.

In the next step, each sentence in the customer reviews is parsed into a dependency tree. A dependency tree describes the structure of a sentence by relating words in terms of binary semantic or syntactic relations [38]. Therefore, each link in the tree explains the relation between two words, which may be close to or far from each other in a sentence. In this methodology, the dependency tree is obtained using PyStanford-Dependencies package in Python [73]. Other than the dependency relation, the parser also provides the part-of-speech for each word. The relevant part-of-speech (POS) tags for the purpose of this methodology are nouns (NN, NNS), proper nouns (NNP, NNPS), and adjective (JJ).

The data preprocessing stage outputs lemmatized sentences, dependency trees of the sentences, and part-of-speech tags of the words in the sentences. As can be seen in Figure 3.2, the lemmatized sentences become the inputs for obtaining the word embedding in Subsection 3.3.2, the trees become the inputs for identifying the related adjective and noun in a sentence in Subsection 3.3.4, while the POS tags become the inputs for various processes in Subsections 3.3.2, 3.3.3, and 3.3.4.

### 3.3.2 Product-feature Words Identification

This subsection is divided into two parts. The first part identifies the product-feature word candidates. It is followed by filtering out the irrelevant product-feature words objectively to obtain the final product-feature words.

### Product-feature Word Candidates Identification

The main input of this stage is a set of lemmatized sentences. In this stage, a word embedding tool word2vec is used to obtain product-feature words. The parameters to be determined for word2vec are the dimensions of the embedding vector, the window size for the context words, the cutoff frequency of words, the usage of either hierarchical softmax or negative sampling [43], and the initial random seed –in order to create a fully replicable result. The word2vec used in this methodology is obtained from `gensim` package in Python [74]. The output from word2vec is the representation of words in real vectors.

The vectors output by word2vec are subsequently clustered with a weighted X-means clustering technique [46]. The weights are required because not all words are equally important. For example, in the customer reviews for laptops, the word "laptop" is arguably more important than "dog", although both words appear in the reviews. In order to reflect the difference in importance, each word is weighted by its tf.idf (term frequency, inverse document frequency). The computed tf.idf are incorporated into X-means clustering as the weight for each word. The formula to compute tf.idf in Equation (3.1) is modified from [75], such that it captures the importance of a word with respect to all documents. In this case, a customer review is considered as a document.

$$ti_i = (tf_i)idf_i = (tf_i)log\frac{|D|}{|d : w_i \in d|} \tag{3.1}$$

where:

$D$ = set of customer reviews; $d \in D$

$idf_i$ = inverse of document frequency of word $i$

$tf_i$ = frequency of word $i$ in the data set

$ti_i$ = tf.idf of word $i$

$w_i$ = the $i$-th word in the vocabulary

Moreover, only nouns are considered as product-feature word candidates, as assumed in the previous literature [59, 64, 51, 40, 56, 61, 67]. Therefore, the words that are not tagged as nouns are excluded from this clustering process.

Based on the X-means clustering result, the word whose vector is the closest to each cluster center is identified as the product-feature word candidates. In order to avoid redundancy caused by highly similar product-feature words, those words are either combined into a phrase or grouped together. For example, the words "heart" and "rate" are combined into a phrase "heartrate"; because the cosine similarity between "heart" and "heartrate", as well as "rate" and "heartrate", is higher than that between "heart" and "rate". Therefore, the phrase "heartrate" is considered as the product-feature word to represent both "heart" and "rate". Furthermore, for the remaining product-feature words, if the cosine similarity between two words are higher than a similarity threshold, then they are still grouped together but not as a phrase, e.g., "web" and "internet" become "web-internet". This approach is taken in order to produce a concise set of product-feature candidates that do not contain words with highly similar meaning.

**Product-feature Words Filtering**

At this stage, a set of product-feature word candidates have been identified. However, the set may contain words that has a high tf.idf value, but not related to product features, e.g., "son". An objective method to filter out such words is proposed here. The input for this method is a set of objective documents of the products, such as the product manuals. In order to avoid bias of overweighting words which are specific to a particular brand of product, it is suggested to select one manual document for one brand of product. Let the proportion of a product-feature word in the manual document $d$ be $p_d$ and the average be $\mu_{p_d}$, for $d =$ 1, 2, ..., $D$. Afterwards, a one-sample t-test is performed with the hypotheses:

Null: $\mu_{p_d} = 0$

Alternative: $\mu_{p_d} > 0$

Based on the hypothesis test, the product-feature words whose proportions are failed to be rejected by the hypothesis test are excluded from the final set of product-feature words. In other words, those product-feature words are statistically uncommon to be found in the manual documents and thus they are considered not related to product features.

Once the final set of product-feature words is obtained, all other words are assigned to the product-feature words based on the highest cosine similarity. As the final adjustment, the assignment of a word is adjusted based on its neighboring words. The underlying assumption for this adjustment is that similar words tend to belong to the same product feature. The process of adjusting the cluster for a word is illustrated with a simplistic 2-dimensional plot in Figure 3.3. It is illustrated that a word (represented as a black dot in Figure 3.3(a)) is initially assigned to product-feature word 1, because it has a higher similarity with product-feature word 1 than product-feature word 2. However, the other words similar to it (represented as white dots in Figure 3.3) are assigned to product-feature word 2. Therefore, the adjustment is made by re-assigning the word from product-feature word 1 to product-feature word 2, as shown in Figure 3.3(b).

The procedure in this section outputs nouns that are the final product-feature words. Furthermore, all nouns discovered in the customer reviews are assigned to the most similar product-feature word. As the

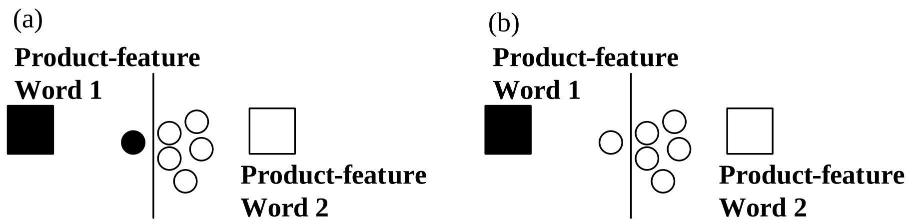



Figure 3.3: Word assignment into clusters: (a) before adjustment, and (b) after adjustment

result, each product-feature word forms a group of words that are similar in meaning with it.

### 3.3.3 Sentiment Intensity Quantification

At this stage, the purpose is to identify sentiment words and quantify their intensity. The inputs are part-of-speech tags, representation of words in real vectors, and a sentiment dictionary.

Based on the part-of-speech tagging result in the Data Preprocessing stage, the adjectives are identified as sentiment words. The sentiment intensity of an adjective is obtained from SenticNet4. Originally, SenticNet is a sentiment dictionary that is developed based on combining ConceptNet and WordNet-Affect [76]. In SenticNet4 [71], a sentiment intensity score is assigned to each concept, such as 0.664 for “good”, 0.179 for “okay”, -0.530 for “faulty”, and -0.900 for “terrible”.

As thorough as it is, there are words that are not included in the SenticNet4 dictionary. The sentiment intensity for each of these words is then obtained by weighted averaging the intensity of the adjectives similar to it, as shown in Equation 3.2. The similar adjectives are identified based on the similarity of the word vectors. The assumption is that similar words, including adjectives, should be embedded close to one another and thus the intensity may be inferred by the surrounding words. This inference makes it possible to obtain sentiment intensity for a new or informal adjective. Therefore, this stage outputs a list of adjectives and their corresponding sentiment intensity scores.

$$int(w) = \sum_{s \in S_w} sim(w,s)(int(s)) \tag{3.2}$$

where:

$int(w)$ = sentiment intensity of word $w$

$S_w$ = set of words that have the highest cosine similarity with word $w$

$sim(w,s)$ = cosine similarity between word $w$ and word $s$

### 3.3.4 Dependency Tree Interpretation

At this stage, the correct connection between a product-feature word (a noun) and a sentiment word (an adjective) in a sentence needs to be discovered using a dependency tree as the main input. The connection is discovered by interpreting the dependency tree as follows: (1) The pairs of adjectives and nouns which are directly connected as parent and child become the output of this stage, (2) The adjective that has no nouns as either its direct parent or child performs further search towards the root of the sentence to discover indirect parents and children. The existence of negation words in a sentence is also important, because it may flip the sentiment expressed towards a product feature. If there is a negation word connected to a pair of adjective and noun, then that pair is marked as having a negation.

The possible connections of adjective and nouns in a dependency tree are illustrated in Figure 3.4. The figure shows the following possible relations: (a) a noun is the direct child of the adjective, (b) a noun is the direct parent of the adjective, (c-1) the adjective has no nouns as either child or parent, (c-2) the adjective moves towards the root and replaces its current direct parent, hence the new shaded box with a "(JJ)" label; the adjective now has a noun as its parent (Indirect Parent), (c3) the adjective moves further towards the root; the adjective now has a noun as its child (Indirect Child). The existence of a negation word negates the relations accordingly, as shown with the bold lines in the figure.

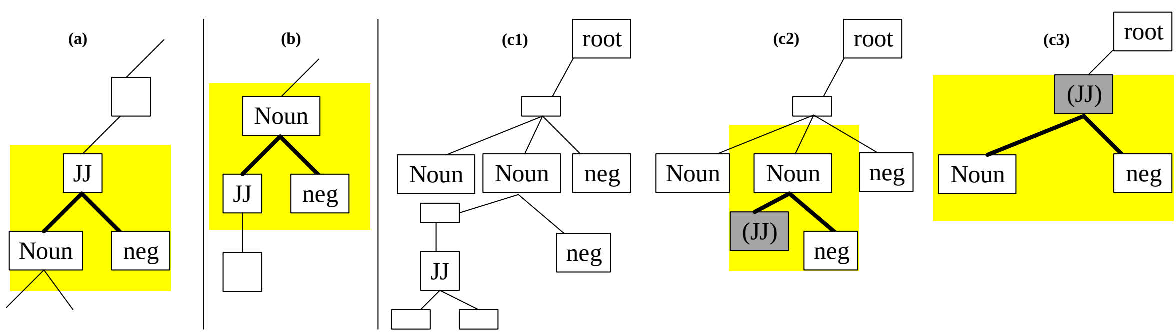



Figure 3.4: Connecting noun to adjective (JJ) in a sentence: (a) Direct Child; (b) Direct Parent; (c-1) no relations found, so the search continues to (c-2) and (c-3) by moving the JJ towards the root; (c-2) Indirect Parent; (c-3) Indirect Child

### 3.3.5 Regression Model Generation

At the last stage, in order to discover variables that are significantly related to sales rank, a linear regression model is used to link all the variables with sales rank. Previously, a linear regression model has been used in

Ref. [49, 77, 61] for the same purpose, assuming a linear relationship between the dependent and independent variables. The assumption is taken because determining the best regression model is beyond the scope of this research. Furthermore, it also depends on the data sets, because different data sets can show different behavior in the relationship.

The dependent variable for the regression model is the log of the sales rank of a product at a particular time. The log of sales rank is justified because, as reported in Ref. [49], the relationship between log sales rank and log sales is close to linear. The aforementioned papers [49, 77, 61] use the log of sales rank as dependent variable. The independent variables are price of the product, as well as the textual and non-textual variables from the reviews. Textual variables are the count of positive or negative comments towards a particular product feature in the reviews. Non-textual variables are the average number of verified purchases, the average star ratings, the average length of reviews, the number of reviews, the percentage of reviews with a good rating (4 and 5 star ratings), and the percentage of reviews with a bad rating (1 and 2 star ratings). As in Ref. [49], the sales rank of the previous day is excluded from independent variables in the regression model. By excluding it, the model reveals more about the relations between review and sales rank. Otherwise, the explanation of variance in sales rank is highly dominated by the sales rank of the previous day. Thus, the regression model can be defined as in Equation (3.3).

$$\begin{aligned} ln(Rank_{i,t}) = Price_{i,t} &+ (aLen_{i,T} + aRat_{i,T} + aVer_{i,T} + nRev_{i,T} + pF_{i,T} + pO_{i,T}) \\ &+ (FP_{f,i,T} + FN_{f,i,T} + tFP_{f,i,T} + tFN_{f,i,T}) + \nu_i \end{aligned} \tag{3.3}$$

where:

$aLen_{i,T}$ = average length of reviews for product $i$ during period $T$

$aRat_{i,T}$ = average rating of reviews for product $i$ during period $T$

$aVer_{i,T}$ = average number of verified purchase of product $i$ during period $T$

$FN_{f,i,T}$ = count of negative comments of feature $f$ for product $i$ during period $T$

$FP_{f,i,T}$ = count of positive comments of feature $f$ for product $i$ during period $T$

$\nu_i$ = fixed-effect variable for product $i$

$nRev_{i,T}$ = number of reviews posted for product $i$ during period $T$

$pF_{i,T}$ = fraction of reviews for product $i$ during period $T$ that are rated 4 and 5 stars

$pO_{i,T}$ = fraction of reviews for product $i$ during period $T$ that are rated 1 and 2 stars

$Price_{i,t}$ = price of item $i$ at time $t$

$Rank_{i,t}$ = rank of product $i$ at time $t$

$tFN_{f,i,T}$ = count of negative comments of feature $f$ in the title review for product $i$ during period $T$

$tFP_{f,i,T}$ = count of positive comments of feature $f$ in the title review for product $i$ during period $T$

The independent variables may correlate to one another. Therefore, stepwise regression is applied in order to avoid highly correlated variables entering the model. Stepwise regression is an algorithm to select a subset of variables in a regression model. The first dependent variable selected into the subset is the one with the highest correlation with the independent variable. The next variable is added into selected set if the ratio of residual sum of squares decrease is greater than an "F-to-enter value. In addition, any variable in the selected set can be dropped if the ratio of residual sum of squares increase is less than an "F-to-drop value. The details of the algorithm can be found in Ref. [78].

For the performance measures of the regression model, two types of R-squared measures are used. The first measure is adjusted R-squared, which provides the percentage of variation in the data explained by the regression model. The value is adjusted with the number of independent variables in the model. The second measure is predicted R-squared. It describes how well the model predicts responses for new observations. This is calculated using the PRESS (predicted residual error sum of squares) statistic and total sum-of-squares (SST) in Equation (3.4). It can be seen that predicted R-squared is a leave-one-out cross-validation technique. If the predicted R-squared is significantly lower than the adjusted R-squared, then the regression model overfits the training data, i.e. the model would not generalize well to a new observation or a new data set.

$$R^2(pred) = \Big(1 - \frac{PRESS}{SST}\Big)100 = \Big(1 - \frac{\sum_{i=1}^{n}(y_i - \hat{y}_{i,-i})^2}{\sum_i (y_i - \bar{y})^2}\Big)100 \tag{3.4}$$

where:

$R^2(pred)$ = predicted R-squared value

$\bar{y}$ = average value of all data's responses

$y_i$ = actual value of the $i$-th data's response

$\hat{y}_{i,-i}$ = predicted value of the $i$-th data's response based on the model that excludes the $i$-th data

## 3.4 Case Study

The methodology proposed in the previous section is applied to two data sets. The data sets correspond to wearable technology products and laptops that have webpages in Amazon.com. They were chosen because wearable technology products were launched about just a decade ago, while laptops have been in the market for a longer time and thus the features have been familiarly known by most people. Furthermore, the consideration for the chosen products is that they need to have an adequate and stable stream of reviews, such that they can be related to the sales rank data. This section describes the data sets, presents examples of results from applying each stage of the methodology, and finally reports the regression results.

The data are accessed from Amazon.com, parsed using `urllib` parser and organized by `beautifulsoup` package in Python. There are 83,565 reviews for wearable technology products and 66,172 for laptops, which were written during the period of January 2015 to February 2017. An example of a review data is shown as follows:

| |
|---|
| Title: "Five Stars", |
| Review: "Great computer. Love it!", |
| ProductName: Acer Aspire E 11 ES1-111M-C40S 11.6-Inch Laptop (Diamond Black), |
| Direct URL: http://www.amazon.com/gp/customer-reviews/R20J53OBD5MTNO/ |
| ref=cm_cr_arp_d_rvw_ttl?ie=UTF8&ASIN=B00MNOPS1C, |
| Month: 02, Year: 2017, Date: 16, Verified: True, Helpful: 0, Rating: 5.0 |

The sales rank data record the periodic sales rank and its corresponding price. For wearable technology products, the data were collected in two periods, i.e. September 2015 to April 2016 and September 2016 to February 2017. At the beginning, there were 140 products whose data were collected. However, in order to keep the ranking consistent in the same category, only items which are ranked in the "Clips, Arms, and Wristbands" category are kept. Furthermore, the duplicated webpages of a product with different sizes or colors are removed, because the webpages share the same reviews. Finally, 35 unique products remain. For laptop products, the data were collected in the periods of October 2015 to June 2016 and November 2016 to February 2017. The collection was started by choosing the laptops listed on the Top 100 and the ranking is recorded according to the "Traditional Laptop" category. Finally, after the removal of discontinued items, 84 products remain in this data set.

### 3.4.1 Processing Review Data

This subsection is divided into two parts, i.e. processing review data to obtain product-feature words and obtaining connections between the product-feature and sentiment words using dependency tree.

### Obtaining Product-feature Words

After being preprocessed, the words from the reviews become the input for word2vec. Since there has been no strict guidelines for determining the parameter values in word2vec, the word2vec parameters are set based on the observations of the preliminary experiment results. For the data set of wearable technology products, the dimensions of the word embedding vector are 100, the window size is 3, the cutoff frequency is 8, hierarchical softmax is used, and the inital random seed is 0. For the data set of laptops, the same set of parameters is used, except the window size is 2. Table 3.5 shows examples of the vector representations of words from laptops dataset. It can be seen that the representations successfully achieve a higher similarity for the pair of similar words (“display” and “screen”) than the other pair (“display” and “storage”).

Table 3.5: Examples of vector representations for selected words, with the cosine similarity with respect to the word ”display”

| Word | d = 1 | d = 2 | . . . | d = 100 | Cosine Similarity |
|---|---|---|---|---|---|
| display | -0.419401556 | 0.673747182 | . . . | -0.773826361 | 1 |
| screen | -0.205376133 | 0.451731592 | . . . | 0.198629543 | 0.65799 |
| storage | -0.443754196 | -0.346733302 | . . . | -1.134292126 | -0.03138 |

In the stage of obtaining product-feature words, X-means clustering is performed by the `pyclustering` package in Python [79] and it outputs cluster centers. The words closest to the centers become the initial product-feature words. After filtering out words that are not related to product features (e.g., “sister”, “son”) and words that are specific to particular brands (e.g., “asus”, “macbook”), the results are shown in Table 3.6 and Table 3.7.

Table 3.6: Product-feature words (wearable technology products)

| Category | Words |
|---|---|
| Final Product-feature Words (15) | activity, alarm, battery, button, charge, clip-strap, company-support-service, data, day, fitness-pal, heart-rate, phone-laptop-app, problem, screen, wristband. |
| Filtered Out Words (8) | bra, money, monitoring, plastic, shade, sister, sleep, yoga. |

Once the final product-feature words have been obtained, all nouns can be assigned to the product-feature words based on the highest cosine similarity. After improving the assignments, according to the adjustment procedure shown in Figure 3.3, the group of words under the same product-feature word becomes more cohesive as presented in Table 3.8 and Table 3.9. The tables display 5 most similar words to the corresponding product-feature words. To highlight the contribution of the adjustment to the cohesive-

Table 3.7: Product-feature words (laptops)

| Category | Words |
|---|---|
| Final Product-feature Words (18) | apps, battery, cable, card, drive, fan, issue, laptop, life, network, office, performance, resolution-quality, screen-display, service, supervisor, track-mouse, web-internet. |
| Filtered Out Words (10) | asus, browsing, casing, cd, everything, facebook, macbook, memory, son, week. |

ness, several movements are provided as examples here, i.e. "device" moves from "phone-laptop-app" to "wristband" cluster in Table 3.8 and "keyboard" moves from "screen-display" to "track-mouse" cluster in Table 3.9. Quantitatively, the adjustment procedure produces a higher average similarity between words within a group. The average cosine similarity between words within a group increases 42% for wearable technology products (from 0.1533 to 0.2176) and 38% for laptops (from 0.1210 to 0.1667).

Table 3.8: Word assignment before and after adjustment for selected product-feature words (wearable technology products)

| Product-feature Word: | phone-laptop-app | | problem | | wristband | |
|---|---|---|---|---|---|---|
| No. | Before | After | Before | After | Before | After |
| 1 | device | app | issue | issue | tracker | device |
| 2 | app | phone | problem | problem | band | band |
| 3 | phone | use | review | reason | one | watch |
| 4 | tool | work | complaint | complaint | watch | wrist |
| 5 | user | apps | motivator | deal | wrist | unit |

Table 3.9: Word assignment before and after adjustment for selected product-feature words (laptops)

| Product-feature Word: | resolution-quality | | screen-display | | track-mouse | |
|---|---|---|---|---|---|---|
| No. | Before | After | Before | After | Before | After |
| 1 | size | size | screen | screen | key | keyboard |
| 2 | quality | quality | keyboard | color | mouse | key |
| 3 | value | speaker | color | display | hp | bit |
| 4 | speaker | sound | display | pad | touchpad | mouse |
| 5 | resolution | resolution | pad | picture | case | reason |

### Obtaining Connections using Dependency Tree

The determination of relations between adjectives and nouns in a sentence relies on a dependency tree. As an example, the dependency tree for the sentence "*however a the construction is plastic it can feel a little cheap in the hand but that shouldn't deter you from purchasing this sleek device a the low price fantastic screen and respectable battery life more than make up for it*" is shown in Figure 3.5(a). As a side note, the errors in the sentence, and the following sentence examples, are caused by the lemmatizer (Section 3.3.1) that mistakenly recognizes "as" as a plural form and thus removes the "s" character from the word. All typographical, grammatical, and other errors are originated from the original reviews.

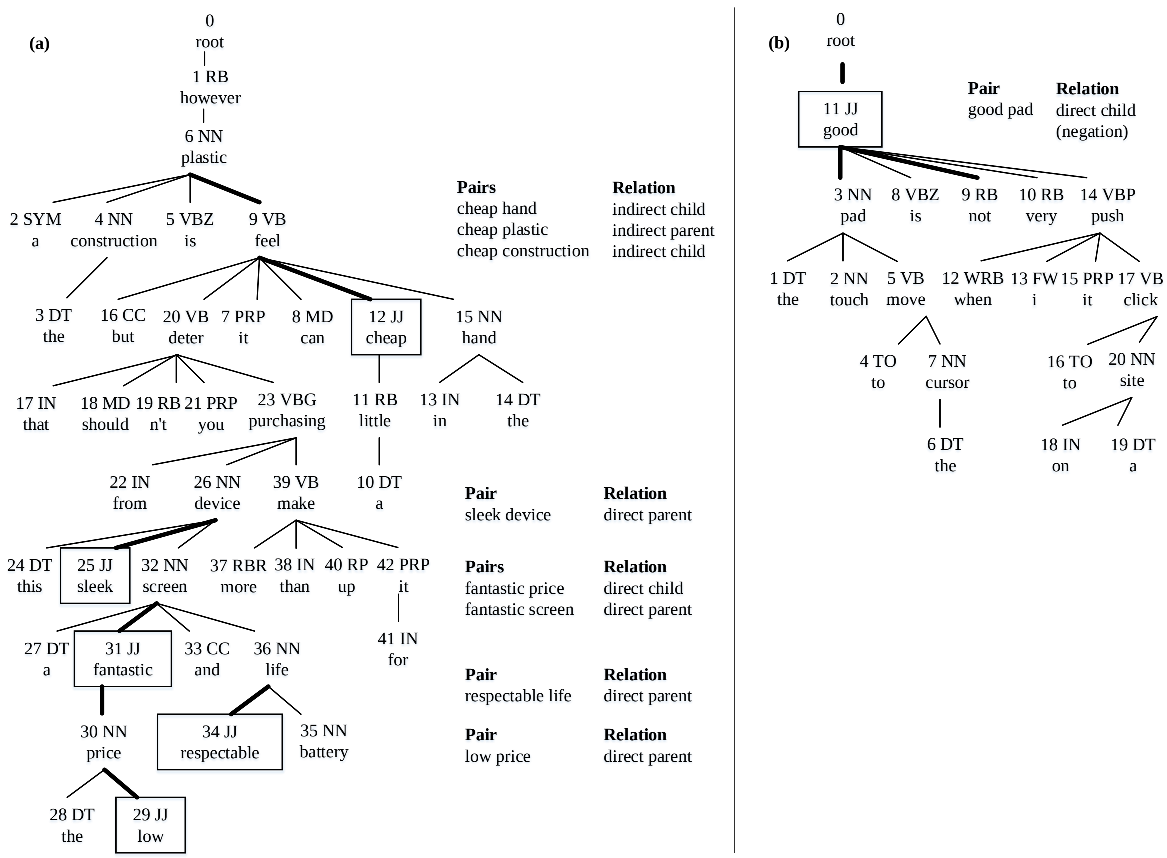



Figure 3.5: Relations between adjectives and nouns in: (a) a sentence without negation, (b) a sentence with negation

In Figure 3.5(a), the direct relations are straightforward, i.e., for the adjectives "sleek", "fantastic", "respectable", and "low". For the adjective "cheap", since it has no nouns as either its direct parent or child, it moves towards the root. As it moves to the position of "feel" (word index 9), it obtains an indirect child "hand" (word index 15) and an indirect parent "plastic" (word index 6). Moving further to the position of “plastic” (word index 6), it obtains an indirect child “construction” (word index 4). Afterwards, moving further until the root of the sentence does not generate any indirect child or parent. This tree becomes an example of various relations that are shown in Figure 3.4. The search for indirect relations brings a trade-off, because it offers the possibility to obtain correct connections, e.g., “cheap” and “construction” in the example, but it is also likely to output false connections, e.g., “cheap” and “hand”. Nevertheless, the indirect pairs are retained in this methodology.

For a sentence that contains a negation word, dependency tree helps to correctly relate the negation with the adjective which it negates. For example, there is a negation in the sentence “*the touch pad to move the cursor is not very good when i push it to click on a site*”. The dependency tree for the sentence is shown in Figure 3.5(b). Based on the tree, it can be determined that the word “not” negates the relation between “good” (word index 11) and “pad” (word index 3). The example in Figure 3.5(b) also presents another advantage of using dependency tree compared to the distance-based approach in Ref. [59, 53], i.e. the word “pad” is not adjacent with the adjective “good” in the sentence, yet the connection is correctly revealed by interpreting the tree.

### 3.4.2 Regression Results

Based on the connections obtained from the dependency tree, each pair of adjective and noun in a review sentence becomes a value that contributes to the corresponding variable in the regression model. The noun is interpreted based on its assignment to a product-feature word and the adjective is quantified based on its sentiment intensity. The existence of negation flips the sign of sentiment intensity. For example, as presented in Figure 3.6, a review sentence for product $i$ at time $t$ that contains “sleek device” contributes as much as 0.853 to the variable “laptop+” for product $i$ at time $t$. The conversion follows the facts that the word “device” is most similar to “laptop”, among all product-feature words, and the word “sleek” has the sentiment intensity of 0.853. In order to reflect the effect of previous days’ reviews towards the sales rank at time $t$, the contribution count is cumulated for the previous $T$ time periods, i.e., $t$, $t-1$, ..., $t-T+1$.

There are 1,990 data points for wearable technology products and 5,587 data points for laptops. Data points included in the regression models must have the sales rank and price recorded for a particular date, as well as having reviews in the period within a week ($T = 7$) from the date. In addition, the reviews must contain identified product-feature words along with the sentiment intensity. The regression analysis is done by applying stepwise regression to eliminate variables that are highly correlated with one another, using $\alpha = 0.05$. For the regression model of wearable technology products, the adjusted R-squared is 84.84% and the predicted R-squared is 84.23%. For laptops, the adjusted R-squared is 70.89% and the predicted R-squared

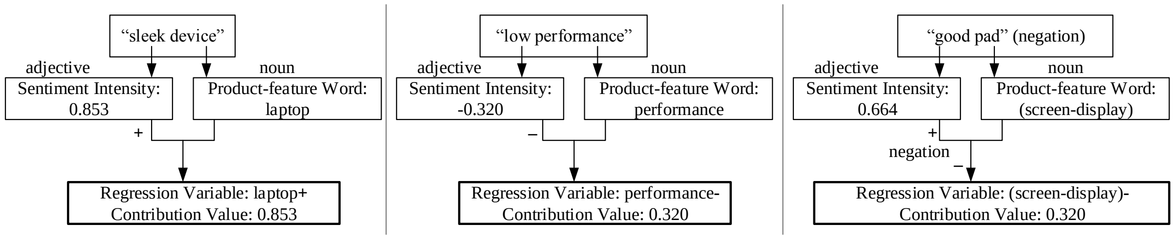



Figure 3.6: Conversion into regression variables

is 70.33%, respectively. The significant independent variables ($\alpha = 0.05$) for both data sets are shown in Table 3.10.

## 3.5 Discussions

This section is divided into two parts. The first part analyzes the variables in the regression results and the second part assesses the sentence interpretation results as well as validating the proposed methodology.

### 3.5.1 Regression Result Analysis

First, it is worth noting from Table 3.10 that many textual-related variables are found to be significantly related to sales rank. It validates the inclusion of textual-related variables in the regression model. Interestingly, for both data sets, there are more significant variables from the review title than from the review content. For the data sets in the case study, this may suggest that a considerable number of customers pay most of their attention towards the review titles, and not reading the review content thoroughly.

Second, the coefficients confirm that the number of reviews and the percentage of reviews with good ratings (4 and 5 stars) are related to better sales rank. Accordingly, the percentage of reviews with bad ratings (1 and 2 stars) and higher price are related to worse sales rank. In the data sets, smaller number indicates better sales rank, i.e. rank 1 is better than rank 2.

Third, among the significant textual-related variables, there are variables whose signs do not follow the common assumption. It is commonly assumed that a positive sentiment about a product feature (e.g., "battery+") is related to better sales rank, and vice versa. However, for example, the variable "activity+" in wearable technology products has a positive coefficient. Further observation reveals that the variable includes not only positive comments about the activity tracker, but also positive comments about doing activity in general, e.g., "*it make me more mindful of the exercise*". The variable "(resolution-quality)+" in laptops also has a positive coefficient. The variable includes comments about sound quality, so terms

Table 3.10: Regression results for wearable technology products and laptops

| Wearable Technology Products | | | Laptops | | |
|---|---|---|---|---|---|
| Variable | Coef | P-Value | Variable | Coef | P-Value |
| Constant | 4.8076 | 0 | Constant | 4.9980 | 0 |
| Price | 0.0048 | 0 | aveFractionVerified | 0.1957 | 0.009 |
| numReviews | -0.0092 | 0 | numReviews | -0.0584 | 0 |
| percent45stars | -0.2359 | 0.002 | percent45stars | -0.3999 | 0 |
| | | | percent12stars | 0.2602 | 0.008 |
| activity+ | 0.0393 | 0.007 | apps- | 0.0724 | 0.001 |
| battery+ | -0.0975 | 0.010 | battery- | 0.0756 | 0.033 |
| charge- | 0.1261 | 0 | drive+ | 0.0702 | 0.021 |
| company-support-service- | -0.0410 | 0 | issue+ | 0.1368 | 0.001 |
| data+ | 0.0421 | 0.031 | laptop- | 0.0430 | 0.008 |
| data- | 0.1603 | 0 | life+ | -0.1102 | 0.001 |
| day- | -0.0661 | 0 | office+ | -0.1966 | 0 |
| heartrate+ | -0.0475 | 0.001 | resolution-quality+ | 0.0440 | 0.009 |
| problem+ | -0.0656 | 0.011 | screen-display+ | -0.1030 | 0 |
| | | | service- | -0.1978 | 0.001 |
| | | | track-mouse+ | -0.1744 | 0 |
| | | | web-internet+ | 0.1748 | 0 |
| | | | web-internet- | 0.0968 | 0.002 |
| title_activity- | -0.2420 | 0.009 | title_apps- | -0.1534 | 0.028 |
| title_alarm- | -0.5140 | 0 | title_battery+ | 0.4199 | 0 |
| title_battery- | -0.6650 | 0 | title_battery- | 0.2676 | 0.003 |
| title_button+ | 1.7050 | 0.010 | title_card+ | -0.2916 | 0 |
| title_button- | 1.0600 | 0 | title_card- | 0.6340 | 0 |
| title_charge- | -0.1917 | 0.002 | title_drive+ | -0.2790 | 0.012 |
| title_company-support-service- | -0.0683 | 0.002 | title_drive- | 0.4530 | 0 |
| title_data- | -0.2760 | 0.008 | title_fan+ | -1.2050 | 0 |
| title_phone-laptop-app- | -0.1399 | 0.010 | title_issue+ | -0.2471 | 0.001 |
| title_problem+ | -0.1489 | 0.009 | title_issue- | -0.1869 | 0.039 |
| title_problem- | -0.1960 | 0 | title_laptop- | -0.2006 | 0 |
| title_screen+ | -0.2300 | 0.034 | title_office- | -1.6980 | 0 |
| title_screen- | -0.7250 | 0 | title_performance+ | -0.0659 | 0.004 |
| title_wristband- | 0.1438 | 0 | title_resolution-quality+ | -0.1204 | 0.004 |
| | | | title_service- | 0.1781 | 0.033 |
| | | | title_supervisor+ | -0.0566 | 0.004 |
| | | | title_supervisor- | 0.1562 | 0 |

such as "right speaker" and "left speaker" appear frequently. Due to the positive sentiment intensity for the adjectives, those neutral terms are interpreted as positive. As the result, it masks the actual complaint about the speaker in a sentence, e.g., "*also my right speaker on the bad doesn't work*". For the "(web-internet)+" variable, which also has a positive coefficient, further observation reveals that most of the sentences are interpreted correctly. However, many positive comments imply that the laptop only serves basic functions for internet, but it does not have capability to do more complicated tasks, e.g., "*perfect for internet use not much else*". Hence, the signs of the regression coefficients that do not follow the common assumption are explained.

An interesting finding is that the variables related to "problem" and "issues" have negative coefficients. It implies that, regardless of the sentiment intensity quantification (e.g., "major problem" is interpreted as "problem+"), the comments about problems are related to better sales rank. Further observation reveals that the word "deal" is assigned into the "problem" product-feature word and it contributes positive terms such as "great deal" and "real deal". Also, the statement of a problem may be followed by the positivity towards the product as a whole, e.g., "*device has a couple issue but is okay especially since it is waterproofed and doesn't require frequent charging*".

In the framework of the five-stage buying decision process, the significant variables in the regression models can suggest the pieces of information that are given significant weights by customers during the Evaluation of Alternatives stage. The information may be used by product designers as one of the inputs to improve product design. From the results shown in Table 3.10, the improvement efforts for wearable technology products may be considered for activity tracking functions, charging process, information presentation, quality and fuctions of the button, and the appearance of the product in general. The improvement efforts for laptops, as shown in Table 3.10, may be considered for nearly all aspects of a laptop, i.e. the applications, battery, storage space and memory, screen resolution, sound quality, and the quality of the laptop in general.

### 3.5.2 Assessment and Validation

To assess the interpretation of sentences from customer reviews, selected sentences from both data sets are presented in Table 3.11. The table provides an example of correctly interpreted sentence and three examples of falsely interpreted sentences for each data set. For the false interpretations, the source of the interpretation inaccuracy is indicated by the numbers inside the parentheses, i.e., (3.2) indicates the non-cohesiveness of the group of words under a product-feature word, (3.3) indicates the inaccuracy of sentiment intensity score assigned to the adjective in the context of the given sentence, and (3.4) indicates the inability to capture the correct relation between an adjective and a noun. Those numbers correspond to the numbers of Methodology

Table 3.11: Assessment of selected preprocessed review sentences

| Assessment | Sentence (wearable technology products) | Adjective-Noun Pair | Regression Variable |
|---|---|---|---|
| True | work well short life span | "short span" | title_battery- |
| False (3.2) | it make me more mindful of the exercise i do during my day | "mindful exercise" | activity+ |
| False (3.3) | dainty feminine long lasting battery | "lasting battery" | title_battery- |
| False (3.4) | work but need better quality control bought 2 only 1 is wearable battery lasted 2 week only | "wearable battery" | title_battery- |
| Assessment | Sentence (laptops) | Adjective-Noun Pair | Regression Variable |
| True | perfect for internet use not much else but based on price it 4 plus star | "perfect internet" | (web-internet)+ |
| False (3.2) | like if yore skyping yo have purple dot all over | "purple dot" | (resolution-quality)+ |
| False (3.3) | also my right speaker on the bad doesnt work | "right speaker" | (resolution-quality)+ |
| False (3.4) | it is hard to get to where i want to go especially on the internet not that it is slow just hard to use | "slow internet" | (web-internet)- |

sub-sections in this chapter.

The ideal validation would be comparing the results from the stages 3.2, 3.3, and 3.4 in Figure 3.2 with a human-annotated corpus for both data sets. However, creating a reliable human-annotated corpus takes a considerable amount of time and effort. Moreover, it is hard to reach agreement between annotators for the tasks in this chapter, e.g., an agreement on the set of relevant product features discussed in the reviews. Nevertheless, since all outputs from those stages build the regression models, the methodology is validated by the performance of the regression models, with predicted R-squared as the performance measure.

The predicted R-squared values are obtained high for both models and they do not drop drastically from the adjusted R-squared values. Thus, it can be concluded that, despite the inaccuracies in the Natural Language Processing, the regression models provide a good description of the relation between reviews and sales rank and they would generalize well to a new data set. It is worth noting that predicting sales rank accurately is not the main purpose of the proposed methodology. Therefore, prediction accuracy is not used as a performance measure.

## 3.6 Conclusion

The chapter proposes a methodology to identify the relation between online customer reviews with sales rank. The methodology consists of five main stages, i.e. data preprocessing, product-feature words identification, sentiment intensity quantification, dependency tree interpretation, and regression model generation. The methodology involves minimal subjective inputs, rules, and decisions, such that the model is objective and generalizable into a new data set. The methodology reveals the product features that are significantly related to sales rank.

The methodology is applied to two data sets, i.e. wearable technology and laptop products. For both data sets, the performance of the regression models is good, i.e. the predicted R-squared is 84.23% for wearable technology products and 70.33% for laptops. The high predicted R-squared values support the claim that the model is generalizable.

Chapter 4

# A Methodology to Construct Customer Choice Sets Using Online Data and Customer Reviews

## 4.1 Introduction

Customer preferences have become an integral part of decision making in engineering design. Recent researches have emphasized the importance of including customer preferences to the decision making process. Li and Azarm [80] apply conjoint analysis to incorporate customer preferences in selecting the best product design. Kumar et al. [81] use nested logit model to accommodate customer preferences in the proposed market-driven product family design methodology. Michalek et al. [82] utilize logit model to model product demand as a part of the product line design optimization. He et al. [83] propose a choice modeling framework for usage context-based design to quantify the impact of usage context towards customer choices. Morrow et al. [84] incorporate a consider-then-choose model into engineering design optimization.

In order to describe customer preferences, an essential component of choice models is choice set. It is defined as a set of product alternatives that are available to a customer [85], who will compare the alternatives before making the final choice [86]. As the choice model is explicitly expressed in terms of product attributes, as well as socio-demographic attributes of customers; high-quality choice set generates a reliable choice model that provides better quality of parameter estimates for the product attributes. Consequently, the choice model would support designers to make better design decisions with respect to customer preferences. Therefore, choice set is also an important factor that supports design decisions in the decision-based design framework [87].

Despite its importance, while the purchase data is generally available, the choice set data is rarely recorded. Wang and Chen [86] propose a method to learn from an existing choice set information in a data set to predict the missing choice sets in another data set. In addition, the customer socio-demographic data also becomes a vital information to generate the prediction of the missing choice sets. Their findings confirm that the learned choice set results in better choice models than both universal and randomly sampled choice sets, in terms of log-likelihood and pseudo R-squared measures.

While the purpose of this chapter is also constructing customer choice sets to create a better choice model, the main contribution of this chapter is proposing a methodology to construct customer choice sets in the absence of both existing choice set and customer socio-demographic data. In the absence of both, the methodology proposes the usage of publicly available online data of product attributes and customer reviews from e-commerce websites. It becomes a promising alternative to conducting survey for collecting customer choice set data, which can be time-consuming, labor-intensive, and expensive [48]. The findings in Section 4.4 show that the usage of online data and customer reviews results in a better choice model compared to the model that uses randomly sampled choice sets. Furthermore, this chapter contributes to linking online self-presentation –which will be discussed in Section 4.2, in the form of customer reviews, with choice modeling. It is achieved by clustering customers based on the reviews, and subsequently utilizing the customer clusters to construct customer choice sets.

The chapter is organized as follows. Section 4.2 discusses relevant researches related to the main topics in this chapter. Section 4.3 elaborates the proposed methodology in constructing customer choice sets using online data and customer reviews, as well as the metric for performance evaluation. Section 4.4 presents the data and results for the case studies. Section 4.5 provides discussion of the findings and limitations of the proposed methodology. Finally, the chapter is concluded in Section 4.6.

## 4.2 Literature Review: Discrete Choice Analysis and Online Self-Presentation

This section presents two main topics related to this chapter. It starts with discussing the discrete choice analysis and the role of choice set in it, then it is followed by the findings from the studies of online self-presentation

### 4.2.1 Discrete Choice Analysis

Discrete Choice Analysis models people's choices among a set of alternatives, i.e., a choice set. It is developed based on the assumption that people act to maximize utility. The utility of alternative $j$ for person $n$ ($U_{nj}$) is formulated as the sum of the observable ($V_{nj}$) and unobservable ($\epsilon_{nj}$) parts of the utility, as shown in Equation 4.1 [85].

$$U_{nj} = V_{nj} + \epsilon_{nj} \tag{4.1}$$

Logit model assumes that the unobservable part is independently and identically distributed as extreme value. Using the assumption, the formula for the probability of person $n$ choosing alternative $i$ ($P_{nj}$) takes a closed form as shown in Equation 4.2 [85]. In the formula, the observable part of the utility is further defined as a linear combination of the alternative's attribute vector $x_{ni}$ and the parameter vector $\boldsymbol{\beta}$. These parameters are estimated by fitting the model to the training data.

$$P_{ni} = \frac{e^{\mathbf{V}_{ni}}}{\sum_j e^{\mathbf{V}_{nj}}} = \frac{e^{\boldsymbol{\beta}'\mathbf{x}_{ni}}}{\sum_j e^{\boldsymbol{\beta}'\mathbf{x}_{nj}}} \tag{4.2}$$

The denominator of Equation 4.2 refers to all alternatives in the choice set. Therefore, the choice probability is directly related to the alternatives included into the choice set. It is often the case that the number of possible alternatives is very large and thus the choice set is constructed randomly [88]. The usage of random choice sets is relatively common in the literature. For example, it is used in the study of warehouse location choice [89], vehicle choice [86], neighborhood selection [90], the benefits of improved water quality in the fishing site [91], and product aesthetics [92]. Therefore, the choice sets that are constructed randomly become the baseline for the performance evaluation in this chapter.

Nevertheless, there are researches suggesting the non-random underlying process of constructing choice set. Gensch [93] proposes a two-stage disaggregate attribute choice model. The model follows a two-stage choice paradigm [94], in which customers filter the set of all feasible alternatives to generate a choice set of few alternatives and closely compares the few alternatives to select one of them. The model requires a survey data, in which customers are asked to rate and rank attributes in each alternative. Wang and Chen [86] proposes a methodology to identify product communities from an existing choice set data (J.D. Power Vehicle Survey) using Newman's modularity method [95] and to obtain customer segmentation from customer socio-demographic profiles using K-means clustering method. The results are used to predict the missing choice sets in another data set of similar products (National Household Travel Survey). In contrast to the aforementioned literature, this chapter proposes a method that does not require survey data of product attribute rating and ranking, existing customer choice set, and customer socio-demographic profiles. Alternatively, in order to construct customer choice sets, the proposed method utilizes product attribute descriptions and customer reviews from a product's webpage.

### 4.2.2 Online Self-Presentation

The emergence of Internet has attracted researchers to study people's self-presentation in the online world. In one of the earliest studies, online personal homepages in Yahoo are successfully classified into one of the five self-presentation strategies that people use in real interpersonal settings, i.e., Ingratiation, Competence, Intimidation, Exemplification, Supplication [96]. In addition, it suggests that gender differences in the real interpersonal settings are reflected in the online homepages.

A more recent study shows that Facebook usage and observable information on a person's Facebook page are associated with personality traits [97]. It implies that real-life personalities are extended into online domain. Similarly, another research identifies that the difference in personal information disclosure is related to the difference in age groups of users [98]. Moreover, the amount of information disclosure also reflects the relationship status of a person.

In terms of people's writings, the word usage in blogs is discovered to be related to the writer's personality to an extent, e.g., Extraversion personality is significantly correlated with the use of positive emotion words [99]. From the study of tweets in Twitter, both semantic and linguistic style features are discovered to be useful to predict personality and profession with high accuracy [100]. It concludes that not only what people say, but how to say it also reveals information about a person's personality and profession.

Although the studies above were not specifically conducted towards customers who write online reviews, there are evidences that online self-presentation represents a person's real characteristics. In conclusion, since the same personality traits and social processes expressed in real life are also expressed in the online world, online interactions have become an extension to people's social lives in the real world [101]. In this chapter, in the absence of the socio-demographic data, customer review is considered as a form of online self-presentations and it is thus used to represent customers.

## 4.3 Methodology

The proposed methodology is summarized in Figure 4.1. It consists of three main parts, i.e. clustering the products, clustering the customers, and finally constructing customer choice sets based on the aforementioned clustering results.

The proposed methodology relies on online data and customer reviews to cluster product and customers. Therefore, the methodology works best when all products in the data set are generally feasible to be purchased by any customer, such that the clustering generates a feasible result as well. However, if there is a hidden constraint –which is not explicitly available on the online data and customer reviews, that strongly restricts a particular customer to a particular subset of products, then the clustering may generate an infeasible result. For example, a customer who would like to purchase an in-car DVD player is strongly restricted to choose from a particular subset of in-car DVD players that is physically and technically compatible with the customer's car. In the online data and customer reviews of the in-car DVD players, however, the compatibility information may not be present. Without revealing the constraint, the product clustering

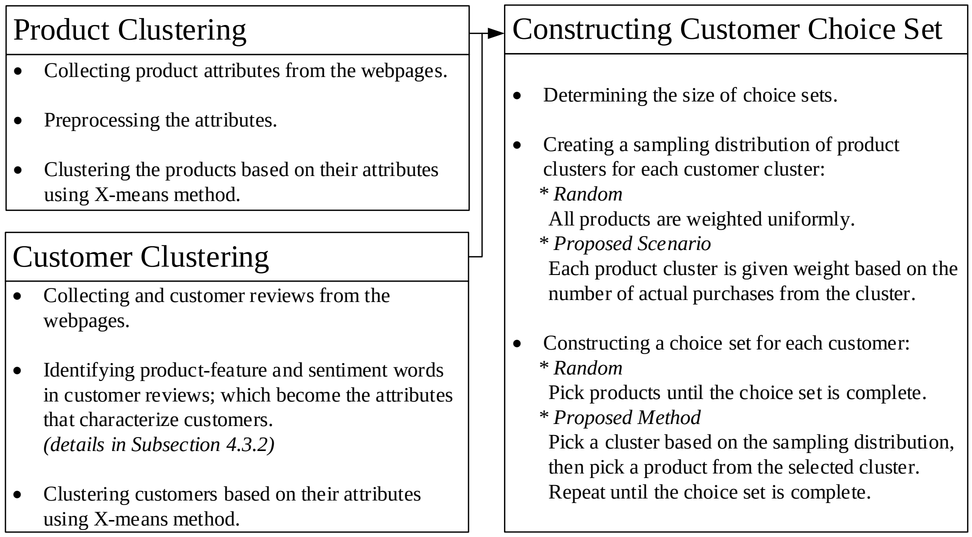



Figure 4.1: Methodology for constructing customer choice sets using online data and customer reviews

might cluster two in-car DVD players in a cluster, despite the fact that they are compatible with different types of car.

### 4.3.1 Clustering Products

In contrast to the existing research that builds product communities based on actual choice set data [86], the proposed methodology clusters the products based on their attributes. The product attributes are acquired from publicly available sources, such as the webpages of products in an e-commerce website. Based on the product attributes, X-means clustering is performed. X-means clustering automatically obtains the best number of clusters by maximizing the Bayesian Information Criterion (BIC) iteratively [46]. It is advantageous compared to the methods that require the number of clusters as an input, such as K-means clustering, because the true number of groups of products is not always known.

Compared to the product communities in [86], product clusters contains a less direct information about the actual customer choice sets. However, the information is proven valuable to construct customer choice sets. As demonstrated later in Section 4.4, the constructed choice sets create choice models that have higher predictive ability than the models that use randomly picked choice sets.

### 4.3.2 Clustering Customers

In contrast to the usage of socio-demographic data to cluster customers in Ref. [86], the proposed methodology utilizes online customer reviews to cluster customers based on the characteristics of their online self-presentations. More specifically, the customers are characterized by the product features that they discuss in the reviews, as well as the sentiment expressed towards those features, e.g., a group of customers who are satisfied with the laptop screen but dissatisfied with the laptop fan. Once each customer has been characterized by a vector that records the frequency of the customer mentioning each product-feature word in the review, then all customers may be clustered using X-means clustering method as well.

There are four stages to identify product-feature and sentiment words from customer reviews based on the methodology in Ref. [35], as shown in Figure 4.2 (Source: [35]). It is considered necessary to summarize each stage in this section, while the details are available in Ref. [35]. The first stage is preprocessing the review data. It involves cleaning the sentences from symbols, lemmatizing the sentences, parsing the sentences into dependency trees, and tagging each word in a sentence by its part-of-speech. In the case studies, since the lemmatization is performed before part-of-speech tagging using the `NLTK` lemmatizer in Python, the lemmatizer mostly replace the plural forms with singular forms. The lemmatizer does not replace the words with their lemmas and, therefore, the subsequent dependency tree parsing and part-of-speech tagging remain accurate.

The second stage is automatically identifying and grouping product-feature words that are discussed in the reviews. This stage is necessary because not all product features that are displayed in a product's webpage are discussed in the reviews, e.g., the sound quality of a laptop's speakers, and vice versa. Moreover, there are similar words that refer to the same product feature, e.g., "screen" and "display", such that they should be interpreted as the same product feature. In order to obtain the product-feature words, a word embedding technique is used to embed the words into real vectors and X-means clustering is used to cluster the word vectors. In order to reflect a word's importance in the clustering process, each word is assigned a weight proportional to its tf.idf (term frequency, inverse document frequency). Based on the cluster centers, the words closest to each center become product-feature word candidates. The word candidates with high similarity are combined into a single entity, e.g., "(web-Internet)".

At this point, further filtering is needed to remove irrelevant words from the product-feature word candidates. The irrelevant words have a high tf.idf, yet are not related to the product itself, e.g., "son"; or too specific for a particular brand, e.g., "ASUS". Those words are filtered out by a t-test that tests the words' average proportions in product manual documents. If the average proportion of those words are not significantly different from zero, the words are considered irrelevant to the product and thus removed from the candidate list. In this research, $\alpha = 5\%$ is used as the significance threshold. The remaining word candidates become the final product-feature words. Finally, to group similar words that refer to the same product feature, all nouns are assigned to the product-feature word that has the highest cosine similarity.

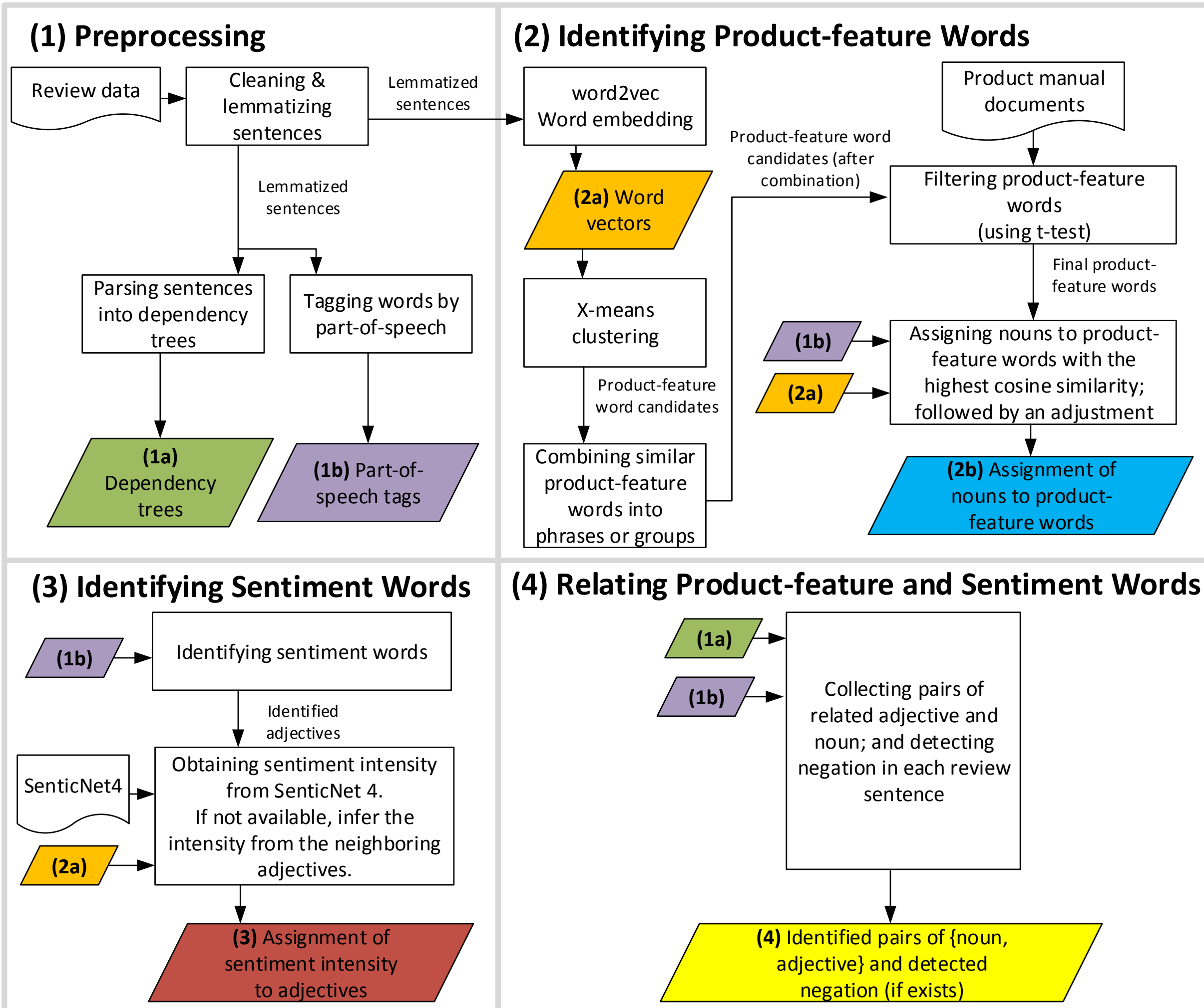



Figure 4.2: Proposed methodology for identifying product-feature and sentiment words in customer reviews

At the second stage, word embedding technique is chosen because it enables the quantification of distance between words, which is useful for grouping similar words. As for the clustering technique, X-means clustering is chosen because the number of product-feature words that are discussed in the customer reviews is not known beforehand; unless the reviews have been manually annotated.

The third stage is identifying sentiment words in the customer review sentences. In this chapter, the identification is done through a word's part-of-speech tag, i.e., an adjective is identified as a sentiment word. Afterwards, the sentiment intensity of those words is obtained from a sentiment dictionary SenticNet4 [71]. The intensity provides the polarity of a sentiment word, i.e., either positive or negative.

Finally, the last stage is relating the results from the previous stages, i.e., product-feature (noun) and

Table 4.1: Center points of product clusters (laptop data set) with the largest number of products, sorted by price

| Aspect | Wang and Chen [86] | This chapter |
|---|---|---|
| Clustering Products | Based on actual choice set data, a network is built using Newman's modularity [95] | Based on product attributes; actual choice set data is unavailable. |
| Choice set data | Available, from J.D. Power Survey | Not available |
| Clustering Customers | Based on actual socio-demographic data | Based on online customer reviews |
| Clustering Method | K-means clustering | X-means clustering |

sentiment (adjective) words in a sentence. This stage is performed using a dependency tree approach because dependency tree may capture the related words regardless of the distance between them. It is advantageous compared to the adjacency-based approach, in which the relation is defined by a fixed window of adjacent words. A pair of adjective and noun is identified to have a relation if the noun is either the direct child or parent of the adjective. If an adjective has no nouns as direct child or parent, it would move towards the root of the sentence. At each step of the move, it would collect the nouns that are now either its parent or child.

After the four stages are performed, each sentence in a customer review may be converted into a list of counts of product-feature words and the corresponding sentiment polarity. The counts are then aggregated for all sentences in a customer review. As the result, each customer is now characterized by a list of counts and it becomes the basis to cluster customers using X-means clustering method; which is chosen because the true number of clusters of customers is not known beforehand.

As a summary of the differences between the research in this chapter and a similar recent work by Wang and Chen [86], Table 4.1 is presented. In particular, the table summarizes the differences in the clustering processes, both in clustering products and clustering customers between the aforementioned researches.

### 4.3.3 Constructing Customer Choice Set

At this point, product clusters and customer clusters have been obtained from the previous subsections. Based on the clustering results, this subsection proposes the scenario to create a probability distribution for sampling the product clusters in order to construct customer choice sets. The reason for creating the probability distribution at cluster level is the absence of the actual choice set data, such that there is not enough confidence to build a probability distribution of products. Moreover, since the number of products is usually large, the inaccuracy of a probability distribution at product level is expected to be higher than it would be at cluster level.

The available data are the actual purchases made by customers and the product clusters. As illustrated

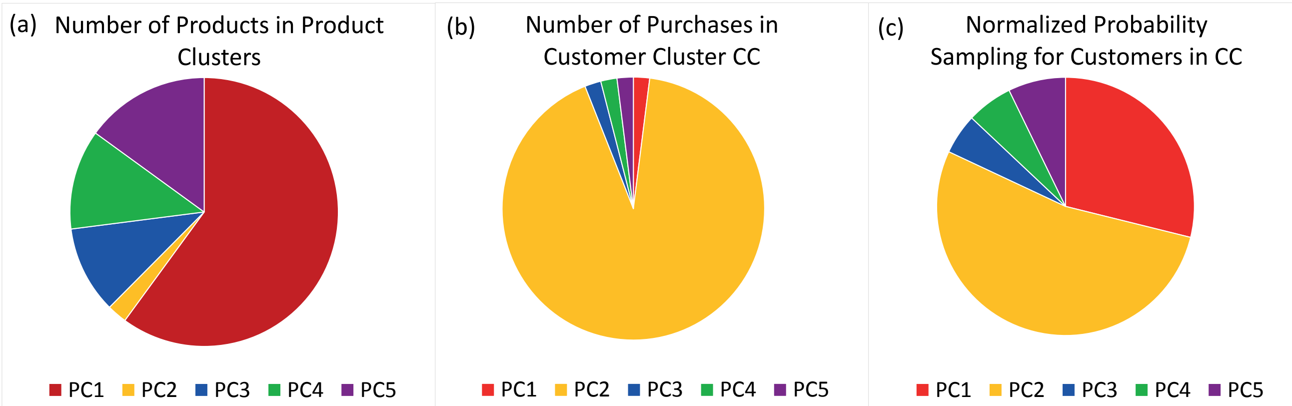



Figure 4.3: Illustration of creating the sampling probability based on the Normalized scenario

in Figure 4.3, the charts represents the size of each product cluster $PC$ (Figure 4.3(a)) and the number of purchases made by customers in a particular customer cluster $CC$ (Figure 4.3(b)). The information from both sources is combined to build the probability distribution of product clusters for each customer cluster.

The proposed scenario called Normalized assigns a probability value to a product cluster $PC$ as a function of the product cluster size and the number of purchases from that product cluster, as defined in Eqn. 4.3 and normalized using Eqn. 4.4 such that the sum equals 1. The first term in Eqn. 4.3 computes the proportion of products in product cluster $PC$ ($I_{PC}$) to the total number of products in all $Q$ clusters. Similarly, the second term computes the proportion of purchased products in product cluster $PC$ made by customers in customer cluster $CC$ ($S_{PC,CC}$) to the total purchases of products in all $Q$ clusters made by customers in customer cluster $CC$. The multiplication of the two terms is denoted as $R_{CC}(PC)$, which is the unnormalized probability of a customer in customer cluster $CC$ to choose a product from product cluster $PC$. In Eqn. 4.4, the normalization results in $P_{CC}(PC)$, i.e., the probability of a customer in customer cluster $CC$ to choose a product from product cluster $PC$ to be included in the choice set, which is illustrated in (Fig. 4.3(c)). The performance of Normalized scenario is compared with Random scenario as the baseline. In Random scenario, a choice set is constructed by picking a set of items randomly.

$$R_{CC}(PC) = \frac{I_{PC}}{\sum_{\forall Q} I_Q} . \frac{S_{PC,CC}}{\sum_{\forall Q} S_{Q,CC}} \tag{4.3}$$

$$P_{CC}(PC) = \frac{R_{CC}(PC)}{\sum_{\forall Q} R_{CC}(Q)} \tag{4.4}$$

Once the choice sets have been constructed for all customers, they become the inputs for the multinomial logit model. As shown in Equation 4.2, each alternative $j$ in a customer's choice set contributes to the denominator of the choice probability formula. The contribution of each alternative is proportional to its utility. In order to define an alternative's utility, there are two functions used in this chapter. The first function, shown in Equation 4.5, defines the utility of alternative $j$ for customer $n$ ($V_{nj}$) as a linear combination of its attributes, i.e., the multiplication of the value of product attribute $k$ of alternative $j$ ($x_{jk}$) and the corresponding logit model parameter for product attribute $k$ ($\beta_k$).

$$V_{nj} = \sum_{k \in K} \beta_k x_{jk} \tag{4.5}$$

The second function, shown in Equation 4.6, defines the utility of alternative $j$ for customer $n$ ($V_{nj}$) by adding an interaction term to the first function. The interaction involves a set of product attributes $K^{Rev}$ that are discussed in customer reviews. It is defined as the multiplication between product attribute $k' \in K^{Rev}$ of alternative $j$ ($x_{jk'}$) and its frequency of being discussed by customer $n$ in the review ($y_{njk'}$) either positively or negatively. Accordingly, the corresponding logit model parameter for the interaction term related to product attribute $k'$ is denoted as $\beta^{Rev}_{k'}$.

$$V_{nj} = \sum_{k \in K} \beta_k x_{jk} + \sum_{k' \in K^{Rev}} \beta^{Rev}_{k'} x_{jk'} y_{njk'} \tag{4.6}$$

### 4.3.4 Performance Evaluation

At this point, the choice sets have been constructed and the utility model has been defined. In order to evaluate the performance of different scenarios in constructing customer choice sets, a data set is divided into a training set and a test set. The training set is used to train the multinomial logit model which provides the estimates of the $\beta$ parameters in the utility function by maximizing the likelihood of the training set. The estimates of the $\beta$ parameters are subsequently applied to predict the probability of purchases of in the test set. In the test set, the choice set for each customer contains all items that have been purchased by customers in both the training and test sets. Therefore, since it is different with the choice sets from either Random or Normalized scenarios, the test set becomes a fair assessment of the predictive ability of the scenario that is used in the training set.

In order to compare the predicted and actual probability distributions, Kullback-Leibler divergence *(KL)* in Equation 4.7 is used as the metric. Kullback-Leibler (KL) divergence measures the difference between two distributions over the same event space [102], such that the higher KL divergence indicates a more different distributions. The actual distribution may be represented by a vector of zeros for all items, except for item $j$ that customer $n$ purchases ($P_{nj}$) that has a value of 1. The prediction on the test set provides the probability of customer $n$ purchasing item $j$ ($Q_{nj}$). A good performance is indicated by the distribution of $Q$ being similar to $P$ and quantified by a low $KL$ value.

$$KL = \sum_j P_{nj} \log \frac{P_{nj}}{Q_{nj}} \tag{4.7}$$

## 4.4 Case Study

In this section, the implementation of the proposed methodology is presented. A data set of laptop products is collected from the website Amazon.com. The data set contains the attributes of 2,631 laptops, which are utilized for clustering products. The data set also contains 46,194 verified reviews from customers who purchased 84 different laptops. The reviews were posted between January 2015 and February 2017 and they are used for clustering customers. In constructing customer choice sets, the customer reviews of products of which the product attributes are inaccessible are excluded. Therefore, the proposed methodology is implemented to a data set of 39,000 customers and 62 products.

At the preprocessing stage for the customer review data, the lemmatizer from `NLTK` package [72] in Python is used to lemmatize the sentences. The Stanford parser from `NLTK` package and `PyStanfordDependencies` package [73] in Python are used to parse each sentence into a dependency tree, as well as tagging each word with its part-of-speech.

### 4.4.1 Product Attributes Data & Product Clustering Result

A product's attributes are collected from its Amazon webpage. For laptops, there is a section that compares similar laptops and lists their attributes, as shown in Figure 4.4 (Source: https://www.amazon.com/dp/B01LZUPUG2, last accessed on May $21^{st}$, 2018). The product attribute information may also be obtained from a product's title and item description section. The attributes are preprocessed such that the unit within an attribute is consistent, e.g., all values in the Processor Speed attribute are converted to have a GHz unit. However, the value itself remains as it is, e.g., the processor speed of 2.3 MHz is converted into 0.0023 GHz; because it is the information displayed and thus received by customers.

The product attributes are used to cluster the products. X-means clustering method is used for the purpose, and it is implemented via `pyclustering` package [79] in Python . There are 25 product clusters obtained, and the top 8 clusters with the highest number of products are shown as the representatives in Table 4.2 with their corresponding center points. The Operating System (OS) is a categorical variable, in which a value of 1 indicates the product uses a Windows system and 0 indicates otherwise. This is a common

| | This item Dell Precision PM7510 15.6-Inch Workstation (Intel Quad Core i5-6300HQ, 256GB SSD, 16GB, 1920x1080, FHD AMD FirePro W5 170 2GB Graphics Windows 10 Pro) (Certified Refurbished) | Acer Aspire E 15 E5-575-33BM 15.6-Inch FHD Notebook (Intel Core i3-7100U 7th Generation, 4GB DDR4, 1TB 5400RPM HD, Intel HD Graphics 620, Windows 10 Home), Obsidian Black | HP Notebook Laptop 15.6 HD Vibrant Display Quad Core AMD E2-7110 APU 1.8GHz 4GB RAM 500GB HDD DVD Windows 10 |
|---|---|---|---|
| | Add to Cart | Add to Cart | Add to Cart |
| **Customer Rating** | (2) | (3616) | (127) |
| **Price** | $1,299.77 | $359.99 | $258.37 |
| **Shipping** | FREE Shipping | FREE Shipping | FREE Shipping |
| **Sold By** | MASTERTRONICS (LIGHTNING FAST SERVICE & SHIPPING) | Amazon.com | VIPOUTLET |
| **RAM Size** | 16 GB | 4 GB | 4 GB |
| **Processor (CPU) Manufacturer** | Intel | Intel | AMD |
| **Processor Speed** | 2.3 MHz | 2.4 GHz | 1.6 GHz |
| **Display Resolution Maximum** | 1920 x 1080 | 1920 x 1080 pixels | 1366 x 768 |
| **Screen Size** | 15.6 in | 15.6 in | 15.6 in |
| **Display Technology** | LED | LED-Lit | LED |
| **Hard-Drive Size** | 256 GB | 1,000 GB | 500 GB |
| **Item Dimensions** | 10.38 x 14.88 x 1.09 in | 15.02 x 10.2 x 1.19 in | 10.04 x 14.37 x 0.39 in |
| **Item Weight** | 6.16 lbs | 5.27 lbs | 5.51 lbs |

Figure 4.4: Snapshot of a similar item section

approach of modeling a categorical variable, as shown in the research by Wang and Chen [86] that models vehicle origin and vehicle type as a sequence of dummy variables that have possible values of 0 and 1. As expected, the clustering result shows that the more expensive laptops generally have higher specifications, as well as being physically larger and heavier.

### 4.4.2 Customer Review Data & Customer Clustering Result

Verified customer reviews are verified by Amazon as being written by customers who have purchased the product. The verification provides the information of the actual purchase made by a particular reviewer. Therefore, for the purpose of the research in this chapter, only verified customer reviews are considered. An example of such review is shown in Figure 4.5 (Source: https://www.amazon.com/gp/customer-reviews/R2LEZTBHDUVOZG/ie=UTF8&ASIN=B00N99FXIS, last accessed on May $21^{st}$, 2018). The sentence is parsed into a dependency tree, as shown in Figure 4.6.

Product-feature words are obtained by applying the word-embedding `gensim` package [74] in Python to obtain the vector representations of the words, then followed by X-means clustering to cluster the vectors. The words closest to the cluster centers are determined as the initial product-feature words. After filtering

Table 4.2: Center points of product clusters (laptop data set) with the largest number of products, sorted by price

| Product Attribute | PC15 | PC22 | PC16 | PC14 | PC0 | PC2 | PC1 | PC6 |
|---|---|---|---|---|---|---|---|---|
| Price ($) | 1791.69 | 1425.62 | 886.62 | 821.87 | 811.75 | 506.98 | 489.02 | 288.36 |
| Processor Speed, PS (GHz) | 2.82 | 2.50 | 2.31 | 2.52 | 1.77 | 2.09 | 2.62 | 1.65 |
| Processor Count, PC | 3.30 | 2.59 | 2.02 | 2.43 | 2.23 | 1.95 | 2.32 | 2.15 |
| Memory (GB) | 20.84 | 14.71 | 9.33 | 10.45 | 9.62 | 5.54 | 8.21 | 3.39 |
| Hard Disk, HD (GB) | 613.76 | 603.42 | 301.82 | 711.63 | 506.68 | 241.49 | 746.44 | 111.65 |
| Ratio (Megapixel/inch) | 0.1333 | 0.4931 | 0.1531 | 0.1328 | 0.0001 | 0.0748 | 0.0672 | 0.0904 |
| Screen Size, SS (in) | 15.55 | 14.82 | 13.56 | 15.58 | 14.47 | 14.01 | 15.53 | 11.60 |
| Volume (in3) | 187.53 | 204.94 | 142.28 | 173.44 | 181.54 | 228.75 | 193.37 | 135.26 |
| Weight (lb) | 4.70 | 4.35 | 3.20 | 4.81 | 3.62 | 3.87 | 21.48 | 2.64 |
| Operating System, OS (1 = Windows) | 1.00 | 0.96 | 0.94 | 0.96 | 0.83 | 0.90 | 0.95 | 0.60 |
| Number of Products | 109 | 138 | 260 | 309 | 684 | 126 | 391 | 164 |

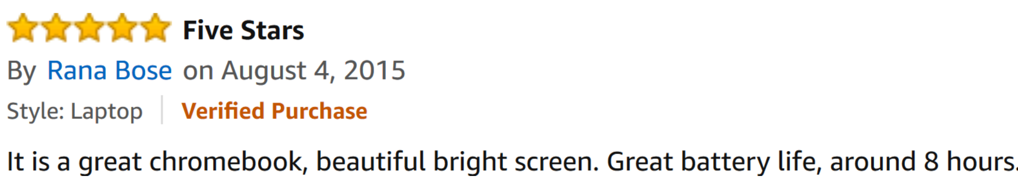


Figure 4.5: A customer review

and grouping similar words, the final product-feature words are shown in Table 4.3. The result is obtained by setting the word2vec parameters as follows: the dimension of the word embedding vector is 100, the window size is 2, the cutoff frequency is 8, hierarchical softmax is used, and the initial random seed is 0.

In Figure 4.6, there are three pairs of adjective and noun identified from the laptop review example, i.e., “great screen”, “beautiful screen”, and “great life”. The word “screen” corresponds to product-feature word “screen-display” and, based on SenticNet4, the polarity of “great” is positive; therefore the first pair is translated into “(screen-display)+”. The second pair is translated into “(screen-display)+” as well, due to

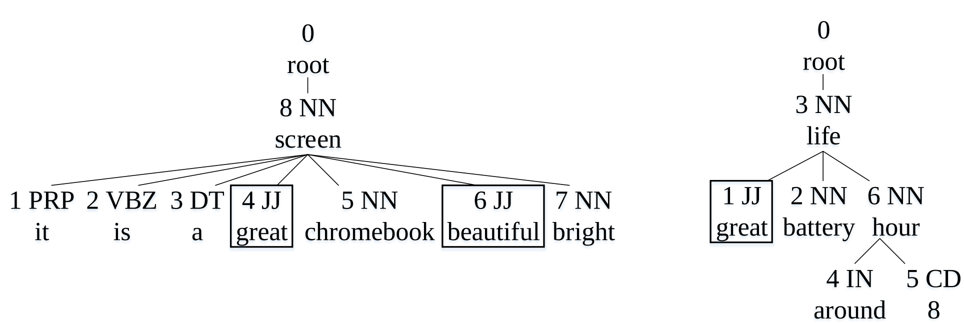


Figure 4.6: The dependency trees of preprocessed sentences in the customer review shown in Figure 4.5

Table 4.3: The 18 product-feature words obtained from the reviews in the Laptops data set

| Data set | Product-feature Words |
|---|---|
| Laptops | apps, battery, cable, card, drive, fan, issue, laptop, life, network, office, performance, resolution-quality, screen-display, service, supervisor, track-mouse, web-Internet. |

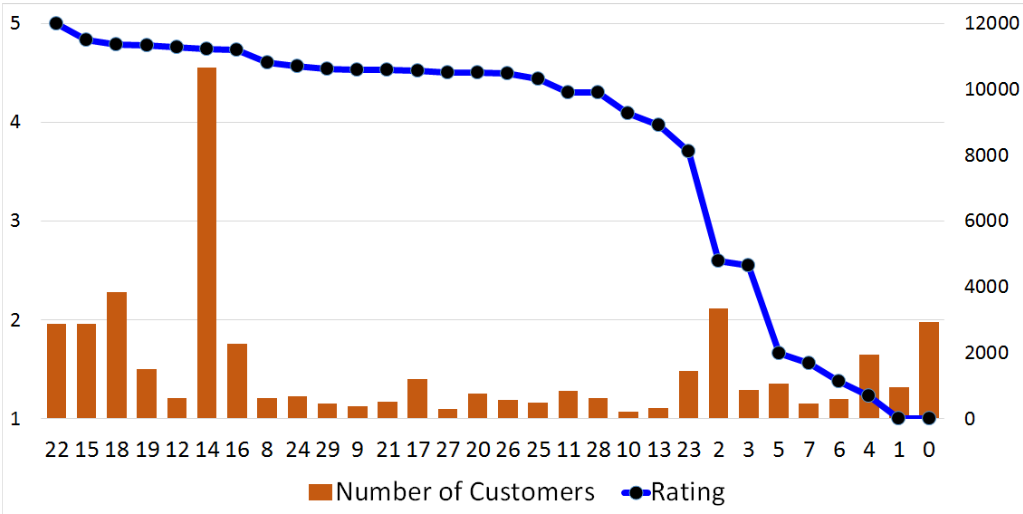



Figure 4.7: Snapshot of customer clusters

the positive polarity of "beautiful". Overall, the review in Figure 4.5 can be converted into a list of counts: "(screen-display)+" = 2, "life+" = 1, and all the remaining pairs are 0.

Based on the counts of product-feature words and the rating assigned to the reviews, customers may be clustered using X-means clustering method. In this case study, each customer is represented by a vector of 37 integers; i.e., 18 product-feature words paired with both positive and negative sentiments, and 1 customer rating. The clustering results in 30 clusters. The number of customers in each cluster as well as each cluster center's rating value are shown in Figure 4.7 for laptop data set. The figure shows that the customer clusters capture the differences among customers, at least based on the cluster's average rating.

The characteristics of each cluster may be analyzed further through the cluster's center. Since Cluster 14 in Figure 4.7 has the highest number of customers, the cluster's characteristics are analyzed here. The center of Cluster 14 is a vector of size 37. Excluding the rating, the remaining 36 values of Cluster 14's center are plotted in Figure 4.8, divided into 18 positive attributes on the left graph and 18 negative attributes on the right graph. The Y-axis of the graphs corresponds to the frequency of a product-feature word and sentiment pair. The center of Cluster 14 is compared to the average of all other clusters' centers, as well as the average of the centers of all other clusters that have ratings of 4 and 5.

It can be observed from Figure 4.8 that customers in Cluster 14, whose average rating is 4.73, are generally

Table 4.4: Comparison between selected sentences from customers in Cluster 14 and Cluster 0

| CC | Sentences |
|---|---|
| 14 | "i have being using it since arrival, the acer has nott disappointed me and i am glad i sold my sell phone to help buy this" (URL: http://www.amazon.com/gp/customer-reviews/RE8QQH55NKO92/)<br><br>"this was purchased for our child in 7th grade, she is very pleased with it, it suits her purpose for school and recreational activities" (URL: http://www.amazon.com/gp/customer-reviews/R31N88N7MGDRKY/) |
| 0 | "very *little* disk **space**, do not buy this laptop, absolutely *terrible* on **space**, not good for saving school work either" (URL: http://www.amazon.com/gp/customer-reviews/R3BAIKVU0E5TA3/)<br><br>"it is a *slow* running **computer** with a *short* battery **life**" (URL: http://www.amazon.com/gp/customer-reviews/R1CCJIS37WSSAX/) |

satisfied customers who write reviews without frequently expressing explicit sentiment towards any product feature in particular. In contrast, compared to the overall average, customers who assigns rating 4 and 5 (excluding Cluster 14) tend to specifically and frequently mention the product features along with their positive or negative sentiment towards them. The examples of original customer reviews, which do not specify explicit sentiment towards any product features, from customers in Cluster 14 are shown in the first row of Table 4.4. In contrast, the sentences from customers in Cluster 0, which explicitly express negative sentiments towards disk space, computer, and battery life; are shown in the second row of Table 4.4.

### 4.4.3 Constructed Customer Choice Set Result

In this chapter, the choice sets are constructed with the choice set sizes of 3, 5, and 7. The numbers are chosen to be relatively small, as suggested by a previous research by Shocker et al. [94] that most customers have a far fewer items than the total number of items available in the market. The suggestion is supported by the extensive research on a large number of product categories by Hauser and Wernerfelt [103], which ranges from table napkins to automobiles, that discovers the average choice set size is between 2 and 8. Table 4.5 presents the number of variables (which may include interaction terms) in the choice model along with the choice set size in the selected papers. Based on Table 4.5, it may be seen that the selection of choice set sizes of 3, 5, and 7 in this chapter for a choice model with 20 variables (10 attributes and 10 interaction terms) is appropriately similar to the sizes that are used in the literature. In this chapter, the varied choice sets are implemented to examine whether or not there are differences in the proposed methodology's performance.

The first product in a choice set is the actual purchase by the customer, which is known from the

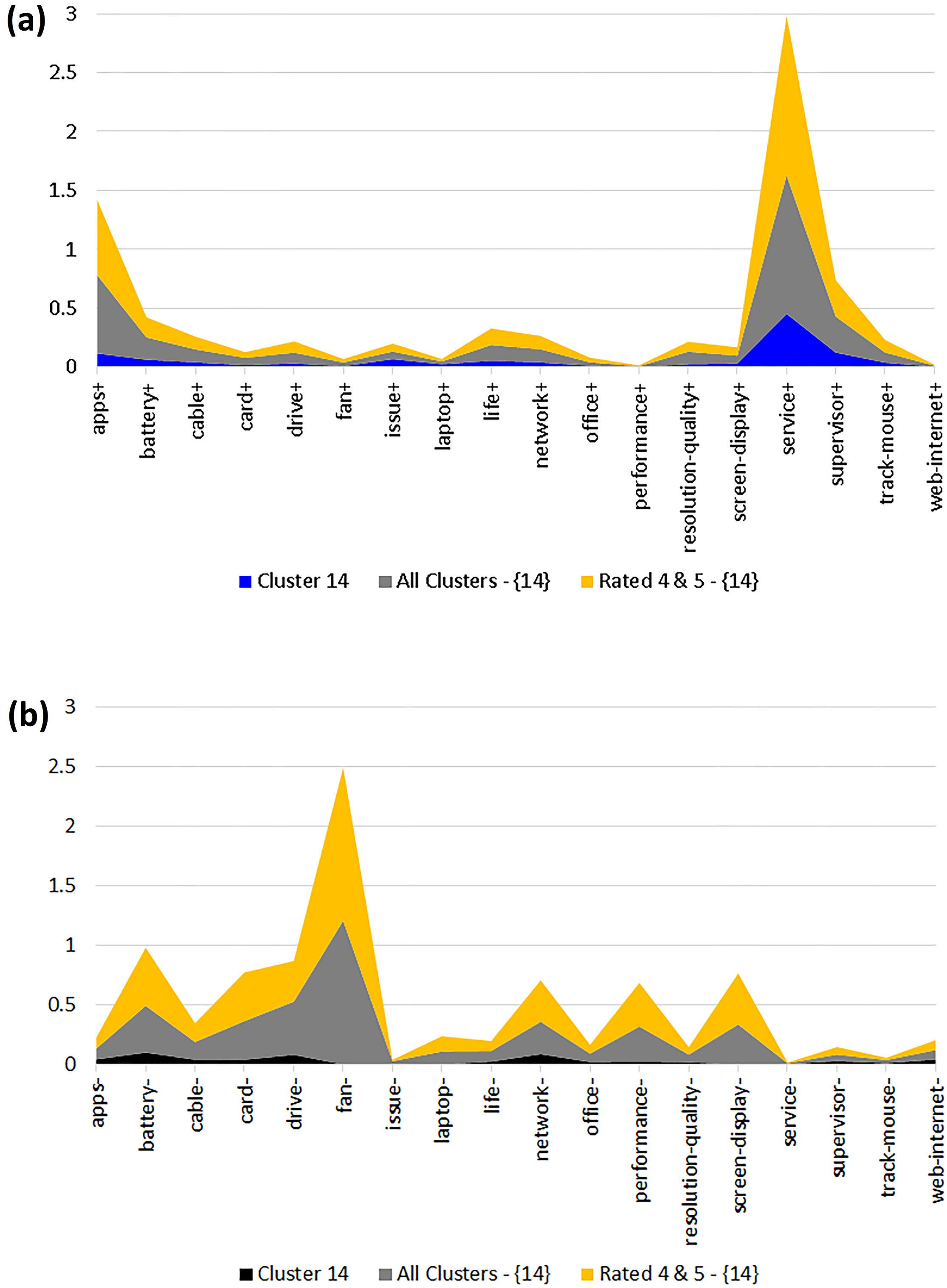



Figure 4.8: Comparison between Cluster 14 with the remaining clusters and with the clusters having 4 and 5 ratings for: (a) positive sentiment and (b) negative sentiment

Table 4.5: Choice set sizes in the literature

| Choice Problem | Number of Variables | Choice Set Size |
|---|---|---|
| Exit selection during building evacuations [104] | 6 | 2 |
| Vehicle selection [86] | 15 | 4 |
| Fishing site selection [91] | 15 | 5 |
| Hotel selection [105] | 16 | 2 |
| Time-period selection in road freight transport [106] | 17 | 2 |
| Mobility-on-demand selection [107] | 21 | 5 |
| Warehouse location selection [89] | 24 | 10 |
| Neighborhood selection [90] | 48 | 11 |

Table 4.6: Example of a customer's constructed choice set

| Choice | Product | Price | PS | PC | Memory | HD | Ratio | SS | Volume | Weight | OS |
|---|---|---|---|---|---|---|---|---|---|---|---|
| Yes | B00N99FXIS | 719.57 | 2.16 | 2 | 4 | 0 | 0.1559 | 13.3 | 80.44 | 2.95 | 0 |
| No | B071XSKHWV | 399.00 | 2.40 | 2 | 6 | 1000 | 0.0672 | 15.6 | 136.78 | 5.20 | 1 |
| No | B015P3SSD2 | 989.99 | 2.60 | 4 | 8 | 1000 | 0.1198 | 17.3 | 262.61 | 8.33 | 1 |
| No | B073R41NPW | 2099.99 | 2.80 | 4 | 32 | 1240 | 0 | 17.3 | 207.22 | 6.17 | 1 |
| No | B06WVGCQ8H | 719.00 | 2.50 | 2 | 12 | 1000 | 0 | 15.6 | 135.00 | 4.80 | 1 |

customer's verified review. The other products to complete the choice set are picked based on either Random or Normalized scenario, with no duplications allowed. An example of the constructed choice set for a customer is shown in Table 4.6. The first column indicates whether or not an item is purchased. The second column shows a product's name and its attributes are shown in the following columns. After constructing choice sets for all customers, the utility of a product for a person may be computed using Equation 4.5. In the formula, K is the set of product attributes, and there are 10 product attributes of laptops as shown in Table 4.6.

For computing the second utility function, defined in Equation 4.6, each product attribute is matched with a product-feature word from the reviews, based on the highest cosine similarity. If a match is found, then the product attribute is included into the set $K^{Rev}$. For example, the word "memory" (one of the product attributes) has the highest similarity with the product-feature word "drive". Therefore *k'* = "memory" is included in $K^{Rev}$. The value of $x_{jk'}$ is the memory (GB) of laptop $j$ and the value of $y_{njk'}$ is the frequency of customer $n$ discussing "memory" in the review, which is represented by the sum of the frequencies of "drive+" and "drive-". In the case that a product attribute does not match with any of the product-feature words, then the attribute is excluded from $K^{Rev}$. The matching between product attributes and product-feature words is shown in Table 4.7.

Table 4.7: Product-feature words with the highest cosine similarity to the product attribute words

| Product Attribute | Product-feature word |
|---|---|
| Price | performance |
| Processor Speed, PS | performance |
| Processor Count, PC | performance |
| Memory | drive |
| Hard Disk, HD | drive |
| Ratio | resolution-quality |
| Screen Size, SS | resolution-quality |
| Volume | laptop |
| Weight | laptop |
| Operating System, OS | apps |

### 4.4.4 Performance Evaluation Result

There are two sets of experiments presented in this subsection. The first set of experiments is used to compare different sampling probability scenarios, i.e., Random and Normalized, with the utility function that only considers product attributes, as shown in Equation 4.5. In the Normalized scenario, the sampling procedure can be done with or without replacement. The with-replacement procedure means that a selected product cluster is returned to the sampling pool, such that it has a chance to be selected again. In both procedures, once a product cluster has been selected, an individual product is subsequently selected randomly from the selected product cluster.

Since probability sampling is involved in constructing customer choice sets, in order to avoid bias due to the random numbers, different starting random seeds are used to construct the choice sets for all customers. In the experiment, 10 starting random seeds are used to create choice set data sets. Each data set is further divided into smaller data sets randomly. In the experiment, each data set is further divided into 10 smaller data sets containing 3,900 customers each; such that finally there are 100 smaller data sets. Each of the smaller data set of size 3,900 becomes the input for training the multinomial logit model, which is implemented via `pylogit` package [108] in Python. The output of the multinomial logit model is a set of coefficients which are the estimates of the $\beta$ parameters in Equation 4.5. The coefficients are subsequently applied to the test data set of size 35,100 to evaluate the predictive ability of the model.

The process is illustrated in Table 4.8. The table contains all items that are purchased by customers in the entire data set along with the values of their attributes, e.g., Processor Speed (PS), Operating System (OS). The *Purchase* column contains the number of purchases of each item, while the *Purchase (Test)* column excludes the purchases in the data that is used to train the multinomial logit model. Based on the *Purchase (Test)* column, the fraction in the *Fraction (Test)* column may be computed and thus represent the actual

Table 4.8: Illustration of comparing true and predicted distributions of purchased item in the test set

| No. | Item | PS | ... | OS | Purchase | Purchase (Test) | Fraction (Test) | Utility | Predicted Fraction |
|---|---|---|---|---|---|---|---|---|---|
| 0 | B00O65HZKS | 2.16 | ... | 1 | 3985 | 3607 | 0.10276353 | 2.56388858 | 0.03111692 |
| 1 | B00NSHLTVG | 2.16 | ... | 1 | 3985 | 3587 | 0.10219373 | 2.62021862 | 0.03292004 |
| 2 | B00O65HZIK | 2.16 | ... | 1 | 3982 | 3560 | 0.10142450 | 2.94748784 | 0.04566585 |
| 3 | B00NSHLUBU | 2.16 | ... | 1 | 3981 | 3569 | 0.10168091 | 2.46594490 | 0.02821371 |
| ... | ... | ... | ... | ... | ... | ... | ... | ... | ... |
| | | | | | $\Sigma = 39000$ | $\Sigma = 35100$ | $\Sigma = 1$ | | $\Sigma = 1$ |

Table 4.9: K-L divergence summary of experiments with different choice set construction scenarios

| Choice Set Size | Scenario | N | Mean | SD | SE Mean |
|---|---|---|---|---|---|
| 3 | Random | 100 | 0.66270 | 0.01660 | 0.00170 |
| | Normalized | 100 | 0.75360 | 0.02830 | 0.00280 |
| | Normalized-replaced | 100 | 0.80930 | 0.02040 | 0.00200 |
| 5 | Random | 100 | 0.65110 | 0.01520 | 0.00150 |
| | Normalized | 100 | 0.60730 | 0.01010 | 0.00100 |
| | Normalized-replaced | 100 | 0.73990 | 0.01260 | 0.00130 |
| 7 | Random | 100 | 0.64870 | 0.01540 | 0.00150 |
| | Normalized | 100 | 0.56178 | 0.00842 | 0.00084 |
| | Normalized-replaced | 100 | 0.70740 | 0.01590 | 0.00160 |

market share in the test set. Based on the utility function in Equation 4.5, the utility may be computed for each item, as shown in the *Utility* column. Finally, the predicted probability of purchasing each item may be computed using Equation 4.2, which may as well be interpreted as the predicted market share, as shown in the last column. The performance metric, KL divergence in Equation 4.7, may then be computed from the columns *Fraction (Test)* and *Predicted Fraction*. The computation of KL divergence in this chapter is implemented via `spacy` package in Python [109].

The performance comparison between choice models that use different choice set construction scenarios is presented in Table 4.9. Based on the average of KL divergence values in 100 samples, the Normalized scenario without replacement procedure is significantly better (p-value = 0.000) than the baseline, i.e., Random scenario, for choice set sizes of 5 and 7. The Normalized scenario with replacement procedure, however, is significantly worse than the baseline for all choice set sizes. Therefore, for the second set of experiments, the Normalized scenario with replacement procedure is excluded.

The second set of experiments is used to compare different sampling probability scenarios, i.e., Random and Normalized, with the utility function that includes the interaction between product attributes and the frequencies of the attributes being discussed in the customer review, as shown in Equation 4.6. The process

Table 4.10: Illustration of the difference in individual utility values towards an item due to the inclusion of the interaction terms in the utility function

| No. | Item | PS | ··· | "perf"$_{n1}$ | "perf"$_{n2}$ | Purchase | Purchase (Test) | Fraction (Test) | Util$_{n1}$ | Util$_{n2}$ |
|---|---|---|---|---|---|---|---|---|---|---|
| 0 | B00O65HZKS | 2.16 | ··· | 1 | 0 | 3985 | 3607 | 0.10276 | $V_{n1,1}$ | $V_{n2,1}$ |
| 1 | B00NSHLTVG | 2.16 | ··· | 1 | 0 | 3985 | 3587 | 0.10219 | $V_{n1,2}$ | $V_{n2,2}$ |
| 2 | B00O65HZIK | 2.16 | ··· | 1 | 0 | 3982 | 3560 | 0.10142 | $V_{n1,3}$ | $V_{n2,3}$ |
| 3 | B00NSHLUBU | 2.16 | ··· | 1 | 0 | 3981 | 3569 | 0.10168 | $V_{n1,4}$ | $V_{n2,4}$ |
| ··· | ··· | ··· | ··· | ··· | ··· | ··· | ··· | ··· | ··· | ··· |

Table 4.11: K-L divergence summary of experiments with different choice set construction scenarios using utility function that includes interaction terms

| Choice Set Size | Scenario | N | Mean | SD | SE Mean |
|---|---|---|---|---|---|
| 3 | Random | 100 | 144546 | 791 | 79 |
| | Normalized | 100 | 147598 | 1088 | 109 |
| 5 | Random | 100 | 144035 | 434 | 43 |
| | Normalized | 100 | 142391 | 356 | 36 |
| 7 | Random | 100 | 143825 | 505 | 51 |
| | Normalized | 100 | 140754 | 317 | 32 |

is illustrated in Table 4.10. The first difference with the illustration in Table 4.8 is the inclusion of the frequency of the product-feature word that is related to a product attribute. For example, Table 4.7 shows that the attribute PS matches with the product-feature word "performance". Customer *n1* discusses it once in the review, while customer *n2* does not discuss it at all; hence the numbers 1 and 0 shown in the columns "$perf$"$_{n1}$ and "$perf$"$_{n2}$. These individual differences cause the utility of each item to differ for each individual, as illustrated by the columns $Utility_{n1}$ and $Utility_{n2}$.

The KL divergence may be computed for an individual by setting $P_{nj}$ in Equation 4.7 equals 1 for item $j$ that is purchased by the individual and 0 for all other items. The total KL divergence of the test set is obtained by summing the KL divergence over all individuals. The comparison between scenarios are shown in Table 4.11. Similar to the result in Table 4.9, the Normalized scenario is significantly better (p-value = 0.000) than the baseline, i.e., Random scenario, for choice set sizes of 5 and 7.

The estimates of $\beta$ parameters for Equation 4.6 that are obtained from the Random and Normalized scenarios are shown in Table 4.12. Those are the coefficients from the data sets of size 3,900 that provides the best (lowest) KL divergence values, i.e., 143,107 (Random scenario with choice set size of 7) and 140,157 (Normalized scenario with choice set size of 7). There is no dramatic difference between scenarios, as the signs of the coefficients of significant variables (p-value $< 0.05$) are the same for both scenarios. The variable *Operating Systems* is significant in the Random scenario, but not in the Normalized. A possible explanation

Table 4.12: Comparison of choice model coefficient estimates between Random and Normalized (Norm) scenarios

| Variable | Coefficient (Random) | SE (Random) | p-value (Random) | Coefficient (Norm) | SE (Norm) | p-value (Norm) |
|---|---|---|---|---|---|---|
| Processor Speed (PS) | 0.0034311 | 0.0060412 | 5.70E-01 | 0.0084258 | 0.0061247 | 1.69E-01 |
| Memory | -0.1099155 | 0.0089848 | **2.06E-34** | -0.1663226 | 0.0105410 | **4.37E-56** |
| Ratio | 0.0000005 | 0.0000001 | **5.34E-05** | 0.0000071 | 0.0000003 | **1.83E-121** |
| Hard Disk (HD) | -0.0026165 | 0.0001005 | **2.19E-149** | -0.0019658 | 0.0000940 | **4.84E-97** |
| Volume | -0.0000034 | 0.0000079 | 6.73E-01 | 0.0000693 | 0.0000838 | 4.09E-01 |
| Weight | 0.0000396 | 0.0005264 | 9.40E-01 | 0.0000071 | 0.0006401 | 9.91E-01 |
| Price | -0.0012553 | 0.0000800 | **1.69E-55** | -0.0014482 | 0.0000876 | **2.29E-61** |
| Processor Count (PC) | -0.2334449 | 0.0230215 | **3.66E-24** | -0.2034307 | 0.0225258 | **1.70E-19** |
| Screen Size (SS) | 0.0766159 | 0.0133647 | **9.88E-09** | 0.2428041 | 0.0151927 | **1.72E-57** |
| Operating System (OS) | -0.3529272 | 0.0527470 | **2.22E-11** | -0.0086996 | 0.0464558 | 8.51E-01 |
| PS*“performance” | 0.1088088 | 0.0429116 | **1.12E-02** | 0.0078080 | 0.0132741 | 5.56E-01 |
| Memory*“drive” | -0.1146116 | 0.0223388 | **2.89E-07** | -0.0596659 | 0.0183426 | **1.14E-03** |
| Ratio*“resolution-quality” | 0.0000023 | 0.0000003 | **2.14E-14** | 0.0000002 | 0.0000003 | 4.93E-01 |
| HD*“drive” | 0.0004134 | 0.0001978 | **3.66E-02** | 0.0002314 | 0.0001793 | 1.97E-01 |
| Volume*“laptop” | -0.0000007 | 0.0000054 | 8.98E-01 | 0.0000866 | 0.0000480 | 7.12E-02 |
| Weight*“laptop” | -0.0000097 | 0.0003648 | 9.79E-01 | -0.0000340 | 0.0005104 | 9.47E-01 |
| Price*“performance” | -0.0001753 | 0.0000945 | 6.35E-02 | -0.0003485 | 0.0001016 | **6.01E-04** |
| PC*“performance” | 0.0125752 | 0.0304176 | 6.79E-01 | 0.0130294 | 0.0296393 | 6.60E-01 |
| SS*“resolution-quality” | -0.0338806 | 0.0177498 | 5.63E-02 | -0.0049395 | 0.0201716 | 8.07E-01 |
| OS*“apps” | 0.0265436 | 0.0535511 | 6.20E-01 | -0.0156893 | 0.0431251 | 7.16E-01 |

is that the Normalized scenario reflects the fact that customers have filtered out the laptops with different operating systems. Therefore, it is no longer significant to predict their choices. The variables such as *Memory*, *Hard Disk*, *Price*, and *Processor Count* have negative coefficients, which means that the increase of these variables is related to the decrease of the probability of being purchased. On the other hand, the increase of *Screen Size* and *Ratio* variables is related to the increase of the purchase probability.

Finally, a comparison is made between the best results from different utility functions, i.e., with or without interaction terms included in the function, as shown in Table 4.13. The KL divergence value for the Normalized scenario in Table 4.9 is converted by computing individual KL divergence first and then summing over all individual, such that the value becomes directly comparable to the Normalized scenario in Table 4.11. The comparison shows that the inclusion of the interaction terms results in a significantly lower (better) (p-value = 0.009) KL divergence.

Table 4.13: Comparison of choice models based on the inclusion of interaction terms in the utility function

| Choice Set Size | Scenario | Interaction Terms | N | Mean | SD | SE Mean |
|---|---|---|---|---|---|---|
| 7 | Normalized | Excluded (Equation 4.5) | 100 | 140855 | 278 | 28 |
| 7 | Normalized | Included (Equation 4.6) | 100 | 140754 | 317 | 32 |

## 4.5 Discussion

The proposed Normalized scenario shown in Figure 4.3 is developed based on two types of information, i.e., the product clusters and the number of purchases within a customer cluster. The multiplication in Equation 4.3 represents the combining of product and customer information. In the Normalized scenario, a product cluster $PC$ obtains a high probability only if it contains many products and customers in $CC$ purchase many products that belong to $PC$. It follows the assumptions that: (1) when there is no additional information, a bigger cluster has a higher probability to be picked; and (2) a cluster of products that is frequently purchased by similar type of customers has a higher probability to be included in those customers' choice sets. The second assumption is parallel to the idea of the neighborhood method in a recommender system [110]. The method may be used to, for example, recommend a movie to a person based on a set of movies that is highly rated by people who like the similar types of movies.

Based on the comparison between Random and Normalized scenarios in Table 4.9, the Normalized scenario without replacement procedure benefits from using the information; i.e., achieving a significantly better predictive ability than the baseline Random scenario, which is indicated by the lower (better) KL divergence values for choice set sizes of 5 and 7. Furthermore, the information is proven valuable because if the information were worthless, then the KL divergence value would not be significantly different with using no information, i.e., the Random scenario.

In both sets of experiments in Section 4.4, the Normalized scenario with the smallest choice set size, i.e., 3, performs worse than the Random scenario. It may be explained that the small choice set size prevents the training set from having enough variation in the selection of items for the choice sets. The small choice set size focuses the selection from the cluster with big probability portions, e.g., PC2 and PC1 in Figure 4.3(c). The test set, however, requires the model to face a highly varied items, because the choice sets include all items in the data set. As the choice set size grows larger, e.g., 5 or 7, it allows the training set to construct choice set by focusing on the clusters with high probabilities, as well as having the opportunity to pick items from clusters with lower probabilities. As the result, the training set has a higher predictive ability on the test set. The similar explanation may be applied to the fact that the replacement procedure results in a significantly worse performance for all choice set sizes, as shown in Table 4.9. The replacement procedure allows a cluster to be chosen repeatedly during the sampling process. Therefore, a cluster with a high probability in the distribution is likely to dominate the constructed choice set in the training set. As the result, the model performs worse when it is applied to predict the test set in the case study.

In the second set of experiments, the utility function in Equation 4.6 is used. It is analogous to the function in a previous research [86], in which customer socio-demographic attributes (e.g., household income, number of children under 18, and fuel price at the vehicle purchase year) are included into the utility function by interacting them with product attributes (e.g., fuel_price * HEV_indicator). In this chapter, since the customer socio-demographic attributes are not available, customer online reviews are utilized to represent the online self-presentation of customers. Specifically, both positive and negative comments from a customer towards particular product features are included in the model, as the comments may indicate the important product features for a customer. The important product features may subsequently be used to characterize customers. Table 4.13 shows that the explicit inclusion of customer reviews into the utility function results in significantly lower (better) KL divergence. The results reaffirm the importance of information from customer reviews in constructing choice models that have a better predictive ability. Moreover, it also shows that customer reviews, as a form of online self-presentation, reflect a persons characteristics to an extent.

As for the limitations, the first limitation of the research comes from the inaccuracy of Natural Language Processing (NLP) tools, which are used to characterize customers based on their reviews. For example, it can be seen in Figure 4.6 that the word "bright" is tagged as a noun (NN), instead of an adjective (JJ). The inaccuracy causes "bright screen" being excluded from the collected pairs of product-feature word and sentiment polarity. The NLP tools with higher accuracy may be expected as there are more annotated data available, as well as due to the advancement of the research in the area. The other limitation is the inaccuracy of product-feature words identification method, as discussed in [35]. It can be seen in Table 4.3 that irrelevant product-feature words appear, e.g., "supervisor". This limitation may be overcome by incorporating manual filtering towards the final product-feature words, which may be performed by a product designer or an expert in the domain. The final limitation is the inability of the multinomial logit model to include non-existing product attributes, although those attributes might have been mentioned by customers in their reviews as an expectation for a product's improvement.

Finally, the challenge of the future research is to discover whether or not online data and customer reviews can replace actual choice set and socio-demographic data, when the latter data are absent. In the scope of the research in this chapter, the claim is that the online data and customer reviews contribute significantly towards constructing choice sets that generate choice models with higher predictive ability compared to constructing choice sets randomly. However, a complete data set that contains customers' purchases, choice sets, socio-demographic data, and those customers' reviews is required in order to answer the question of replacing the actual data with online data.

## 4.6 Conclusion

In the absence of actual choice set and socio-demographic data, the publicly available online data of product attributes and customer reviews are valuable to construct customer choice sets. In the proposed Normalized scenario, the information is utilized to build a probability sampling for constructing customer choice sets.

In the case study, the constructed choice sets generate choice models with significantly higher predictive ability compared to the models that are created using Random scenario. Furthermore, the explicit inclusion of customer reviews to the utility function results in choice models with significantly higher predictive ability. Since the choice models with higher predictive ability provide more accurate parameter estimates of the product attribute variables, they become more useful to support designers in making engineering design decisions, especially by allowing designers to observe the change in demand with respect to the changes in the product attributes.

## Chapter 5

# A Methodology to Identify Product Usage Contexts from Online Customer Reviews

## 5.1 Introduction

Product usage context is one of the important factors that affect product design and beyond. Green et al. [111], in a study of the products that perform the primary function to broadcast light and allow mobility, conclude that the differences in product requirement design targets and customer needs may be convincingly explained according to the differences in product usage contexts. Green et al. [112], in a study of food boiling and mobile lighting products, conduct a survey that indicates different product preferences for different usage contexts. Due to its relevance towards product design, Kanis [113] states that, instead of relying on assumed usage, discovering the actual usage context is indisputably critical.

The importance of product usage context extends beyond product design. In an earlier research, Belk [114] shows indication that consumer behavior is influenced by situational characteristics, including the Task Definition characteristic. Ram and Jung [115] show statistically significant differences in consumer satisfaction among groups with different usage contexts, i.e., usage frequency, usage function, and usage situation. He et al. [83] argue that the reasons behind and the situations under which a product is being used, i.e., usage contexts, are essential to fully understand and model heterogeneous choice behavior. Related to choice behavior, Ratneshwar and Shocker [116] theorize that usage contexts act as environmental constraints that help define consumers' goals, such that they limit the nature of products that may be chosen to achieve those goals.

Considering the importance of product usage contexts, it is beneficial to understand product usage contexts. There are at least three benefits from understanding product usage contexts [111]:

1. Facilitate and organize the customer needs gathering process more effectively.
2. Improve the task of setting target design values, by taken usage contexts into consideration.
3. Leverage the known to design for the unknown. The contextual understanding has been shown to improve the final designs, even when the design problems are outside of the designer's expertise [117].

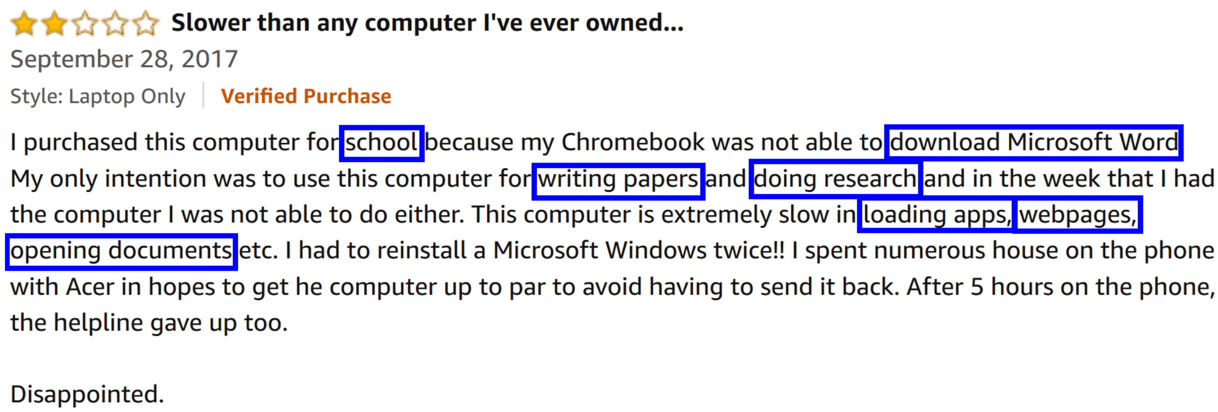
In the literature, product usage context data are mostly collected through survey-based methods and the list of usage contexts has been predetermined, as discussed in Subsection 5.2.1. The main disadvantage of survey-based methods is that they may be expensive and time consuming to conduct [118, 119]. As an alternative, online customer reviews are publicly available data that may be utilized for the purpose. Online customer reviews are mostly written based on the willingness of customers out of their own interests [119]. Customers intentionally and voluntarily invest time and energy into sharing their opinions in their reviews, such that a high level of authenticity may be expected [48]. It implies that, in terms of product usage contexts, the usage contexts that are mentioned in the reviews are of customers' true interests.

The massive volume of customer reviews, however, makes it virtually impossible to analyze the reviews manually. Therefore, this chapter proposes a methodology to automatically identify product usage contexts from online customer reviews, using as little supervision as possible. In order to achieve that purpose, the proposed data-driven methodology in this chapter is supported by machine learning and Natural Language Processing tools.

For customers and e-commerce websites, this chapter contributes in proposing the possibility to allow customers to filter products based on their prioritized usage contexts. In the laptop category, up to May 2, 2019, both Amazon.com and BestBuy.com only offer the usage-related filtering by three general groups, i.e., Personal, Business, and Gaming. These three groups may not represent a customer's prioritized usage contexts well. Moreover, this chapter shows that the overall rating may not always strongly correlate with the sentiment towards a particular usage context, i.e., a high overall rating may not guarantee that a product is good for a particular usage context.

For product designers, this chapter contributes in providing an insight of customer sentiment towards the usage contexts that the product is either intentionally or not intentionally designed for. For example, a review for a laptop that is marketed with the slogan "Better Everyday Computing" is shown in Figure 5.1 (Source: https://www.amazon.com/gp/customer-reviews/R3HWIC4CAWWVJ8?ASIN=B01K1IO3QW). The rectangles in the figure highlight the usage contexts in that customer review. Based on the review, it is obvious that the customer is not satisfied with the laptop's performance in several usage contexts including writing papers. In other words, the reviewer's aspect sentiment towards the aspect of "writing papers" is negative. Considering the volume of the reviews for a product which is commonly in the order of thousands, the proposed methodology significantly helps designers to focus on several specific reviews regarding particular usage contexts, which may or may not have been previously realized by the designers.

The chapter is organized as follows. Section 5.2 discusses the literature that is related to product usage contexts and recent literature in data-driven approach to product design, especially for identifying usage



**Slower than any computer I've ever owned...**

September 28, 2017

Style: Laptop Only | Verified Purchase

I purchased this computer for school because my Chromebook was not able to download Microsoft Word My only intention was to use this computer for writing papers and doing research and in the week that I had the computer I was not able to do either. This computer is extremely slow in loading apps, webpages, opening documents etc. I had to reinstall a Microsoft Windows twice!! I spent numerous house on the phone with Acer in hopes to get he computer up to par to avoid having to send it back. After 5 hours on the phone, the helpline gave up too.

Disappointed.

Figure 5.1: An example of customer review that perceives a laptop negatively in the usage context of writing papers

contexts. The section also provides a review on neural network classifier and an Attention-based Long Short Term Memory Network, which is used for analyzing the aspect sentiment. Section 5.3 presents each stage of the methodology. Section 5.4 shows the results of implementing the methodology to the data sets of laptops and tablets. Section 5.5 discusses performance of the proposed methodology, as well as providing further discussion on the chapter's contributions for customers and designers. Finally, Section 5.6 concludes this chapter.

## 5.2 Literature Review: Usage Contexts and Neural Network

This section presents the definitions of product usage contexts and the previous works in identifying the contexts. The second subsection specifically discusses the comparison between the research in this chapter and relevant recent papers in identifying usage contexts from online reviews. The remaining subsections provides the review on neural networks.

### 5.2.1 Product Usage Contexts

LaFleur [120] defines four environments in the design engineering framework, i.e., application, design, verification, and construction. The environment that is related to the product usage context is the application environment. Application environment is defined as the actual situation that a device encounters, including conditions, constraints, and actual tasks to perform. Ram and Jung [121] state that the usage of a product may be examined from three perspectives, i.e., social interaction, experiential consumption, and functional utilization. The functional utilization perspective studies the usage of product attributes in different situations. In particular, for technological products such as personal computers, customers may use a combination of features or functions in order to enjoy usage variety in different applications, e.g., word-processing, computer games, etc. The variety results from both the product attributes and the usage situations. Green et al. [111] define product usage context as all factors relating to the situation in which a product may be used, including how the product is used (for what application). Finally, He et al. [83] define product usage context as "*all aspects describing the context of product use that vary under different use conditions and affect product performance and/or consumer preferences for the product attributes.*" Based on the definitions from the literature, *the product usage context in this chapter includes the tasks or applications that a user performs using the product.*

There have been works in the literature that collect data regarding product usage contexts. In the study of the usage contexts of videocasette recorder (VCR), computer, microwave, and food processor, the data are collected from self-reported questionnaires and diaries [121]. Similarly, a field survey is conducted in order to study the usage context of VCR [115]. In the study of choice modeling for usage context-based design, the usage context data are collected from the combination of surveying respondent and secondary data [83]. More recently, in the study of automatically identifying usage context using Convolutional Neural Network, the data are collected from the accelerometer and gyroscope, which are embedded in the smartphones that are attached to the respondents [122]. Zhou et al. [62] utilize the usage contexts in order to elicit latent customer needs from customer reviews. However, the use case categories are subjectively predetermined (e.g., Contextual Events use case category includes "Seated", "On a trip", "Cooking", and "Working out") as shown in Figure 5.2 (Source: [62]), instead of being identified from the customer reviews. As the consequence, it requires either an expert in the product domain or reading a lot of customer reviews to create a reasonable set of use case categories. All other aforementioned works in Refs. [121, 115, 83, 122] also predetermine the usage contexts subjectively. *In contrast to the aforementioned works, this research uses publicly available online customer reviews as the data to automatically identify product usage contexts.*

### 5.2.2 Usage Context Research in Data-driven Product Design Domain

From the discussion about recent research of data-driven approaches to product design in Chapter 2, it may be seen that online customer reviews have been utilized to support the product development process in various ways. Nevertheless, the idea of identifying usage contexts using online customer reviews has not been extensively explored. In fact, in the comprehensive review on recent advances in data-driven product design [5], utilizing the Big Data to reveal product usage contexts is mentioned as one of the several crucial challenges and open problems in the product design domain.

In the research that is related to identifying usage contexts from online reviews, a recently published

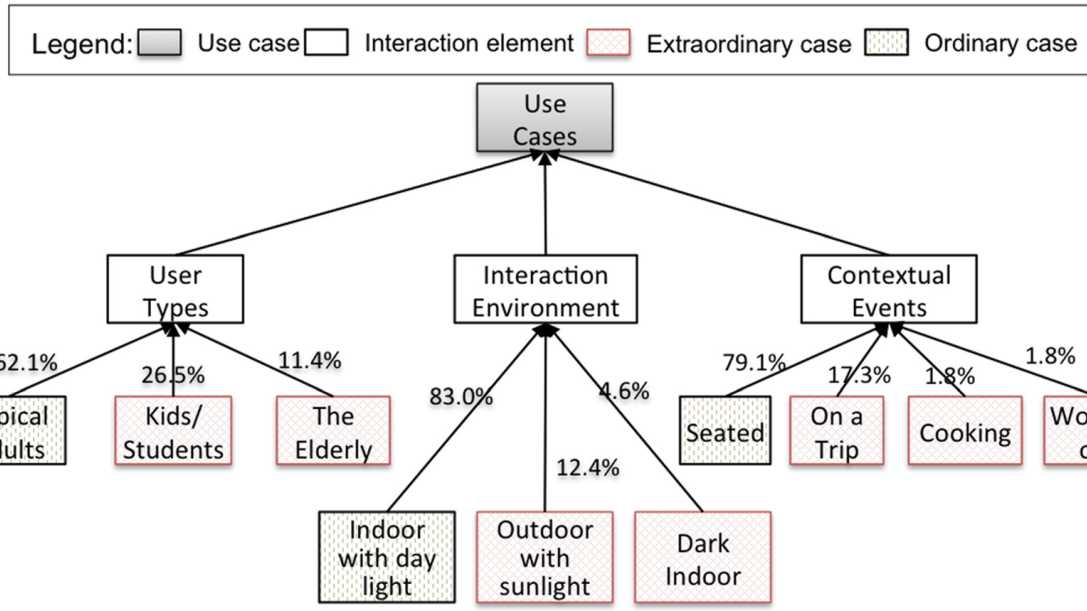



Figure 5.2: Example of use cases of tablets

work by Yang et al. [123] addresses the challenge by proposing a faceted model of user experience. The model is illustrated in Fig 5.3 (Source: [123]). Referring to the model, the usage context in this chapter is represented by the sub-facet "Activities" in the Situation Facet.

Despite the similarity in the attempt to identify the activities in the Situation Facet, there are at least three main differences between the research in this chapter and the work done by Yang et al. [123], i.e., the methodology, the level of generalization and automation, and the application of the obtained knowledge, as summarized in Table 5.1 which also includes the aforementioned work by Zhou et al. [62]. The differences are further elaborated below.

First, there are two differences in the methodology as follows:

1. Yang et al. [123] identify product, situation, and sentiment facets separately. Those facets are subsequently combined without considering the relations between words in a sentence. Consequently, the result may be partially accurate, as shown by the examples from the customer reviews of a laptop below. In the examples, the Situation Facets are obtained from the result of the case study in Yang et al. [123] and the Product Facets are inferred from the same source.

   (a) Sentence: "i've had the laptop *for a day* - i'm pretty disappointed that the 450 g2 does not have a removable *battery*, and uses a different power supply plug than the 650 g1"

   Product Facet: battery; Situation Facet: for a day; User Sentiment State: negative

   Comment: The triple of product, situation, and sentiment may be interpreted as the battery

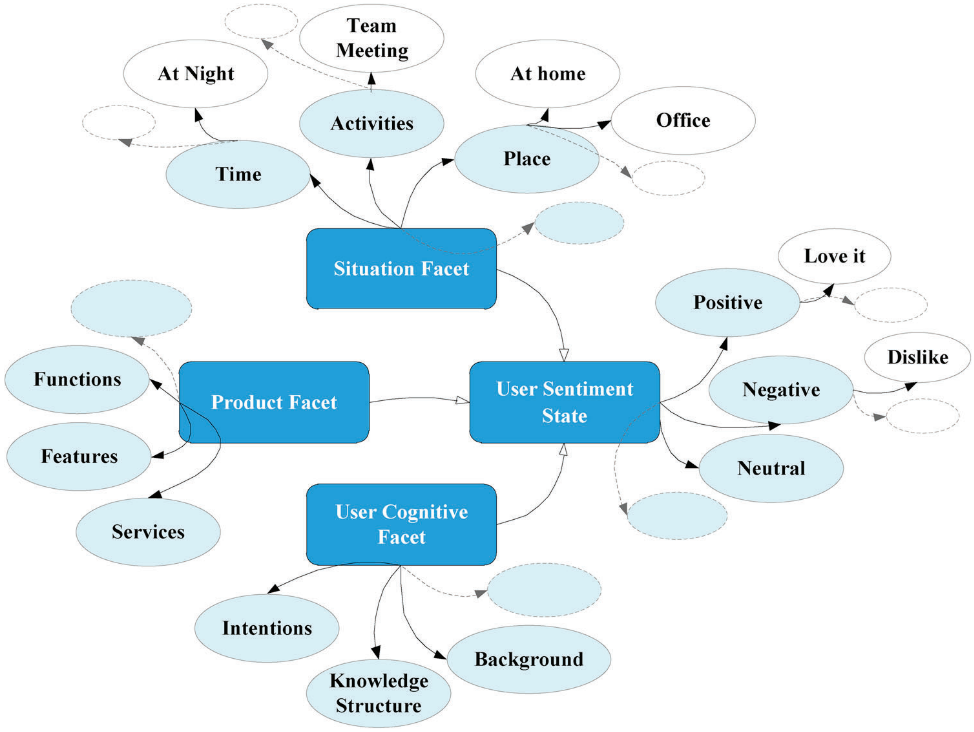



Figure 5.3: A faceted model of user experience

lasting for a day and it is perceived negatively by the customer.

(b) Sentence: "casual and hardcore gamers will find a lot to love here , and video or *photo* editing folks ( who don't need to rely on a laptop screen for color accuracy ) will feel *at home*"
Product Facet: photo; Situation Facet: at home; User Sentiment State: positive
Comment: The triple of product, situation, and sentiment may be interpreted as a positive experience of using a photo-related feature at home, although the term "at home" in this sentence has a different word sense.

*Therefore, to avoid the partially accurate results due to combining separately identified facets from a sentence, the proposed methodology in this chapter attempts to identify the usage contexts along with their corresponding aspect sentiments.* Based on the approach in this chapter, the sentence in (b) will produce "photo editing" as a specific usage context, as opposed to just "photo" that may refer to different usage contexts (e.g., taking photo, storing photo, photo editing, etc.) and therefore may require a designer to read the entire sentence in order to clarify it.

2. Zhou et al. [62] and Yang et al. [123] infer the sentiment at the sentence level. Consequently, the

Table 5.1: The summary of differences between the relevant recent works and this research

| Yang et al. (2019) | Zhou et al. (2015) | This research |
|---|---|---|
| **Method** | | |
| • Identify Product Facet, Situation Facet, and User Sentiment State separately.<br>• Sentiment analysis: sentence sentiment. | • Determine the use cases subjectively.<br>• Sentence sentiment. | • Identify the usage contexts along with their corresponding aspect sentiments.<br>• Aspect sentiment. |
| **Level of Generalization or Automation** | | |
| • Situation Facet: sentences without product-feature or opinion words are discarded. | • Use cases are predetermined; not utilizing review sentences. | • Sentences without product-feature or opinion words are not discarded. |
| • Situation Facet: sentences are filtered using a model that requires manually labeled and domain-specific seeds. | • Use cases are predetermined; not utilizing review sentences. | • Sentence classifier are trained using the training set that is constructed based on domain-free grammatical rules. |
| • Sentiment: based on part-of-speech tagging, top *k* positive and negative words are chosen as seeds for sentiment analysis. | • Fuzzy Support Vector Machines are trained by lexicons of sentiment words. | • Aspect sentiment analysis model is trained by a big corpus. |
| • Clustering: local and global connection scores become the basis for clustering; the weights of each connection are subjective and the similarity measure to compute global connection is not mentioned. | • The clusters of use cases have been predetermined subjectively. | • Word vectors become the basis for clustering; they capture the meaning between phrases beyond the sameness of words. |
| • Clustering: predetermined *k* in k-nearest neighbors. | • Predetermined. | • X-means clustering determines *k* automatically. |
| **Application of Obtained Knowledge** | | |
| • Building network of Product & Situation Facets to explain User Sentiment | • Identifying extraordinary use cases and the latent needs. | • Identifying a product's position in the market.<br>• Filtering products by usage contexts. |

obtained sentiment may not actually refer to a particular usage context in the sentence. On the other hand, the state-of-the-art sentiment analysis has been performed at the aspect level, because an aspect is an integral part of an opinion. An opinion is defined as a quintuple of an entity, an aspect of the entity, the orientation of the opinion about the aspect, the opinion holder, and the time when the opinion is expressed [124]. In the context of customer reviews, aspects are defined as opinion targets, i.e., the specific features of a product or service that the reviewer likes or dislikes [125]. Thus, aspect sentiment analysis is defined as a task to determine whether an opinion on an aspect is positive, neutral, or negative [124]. Identifying aspect sentiment is crucial because a sentence may express opposite polarities about different aspects of a product [125], as shown by the following sentence: "The voice of my Moto phone was unclear, but the camera was good" [124]. *Therefore, to obtain the corresponding sentiment towards the usage contexts, this chapter applies the aspect sentiment analysis.*

Regarding the level of generalization and automation, it is argued here that a number of predetermined or subjective inputs in the methodologies proposed by Zhou et al. [62] and Yang et al. [123] may hinder their abilities to generalize to other domain of products, since it is dependent upon the subjective inputs from the experts in a particular domain. In Ref. [123], the subjective inputs are as follows:

1. In identifying Situation Facet, the sentences without product-feature or opinion words (i.e., adjectives or verbs that contain sentiment) are discarded. On the other hand, the methodology in this chapter does not rely on the existence of both product-feature and opinion words. In fact, the sentences without

sentiments are still useful to inform product designers about customers' usage contexts, regardless of the existence of the sentiment. For example, the information may be utilized to obtain extraordinary usage contexts and identify lead users [62]. *Therefore, in this chapter, the following sentences are not discarded and the usage contexts (in italic font) are successfully identified*:

   (a) "this laptop will get the job done : *writing papers* , youtube videos in 720 ( anything above 720 will have issues ) , gaming here and there ( i can play league of legends with 30 - 60 fps ) , etc" (Note: no opinion words)

   (b) "this isn't for *hardcore gaming*" (Note: no product-feature and opinion words)

2. In identifying Situation Facet, the sentences are filtered by a model that requires initial seeds that are manually labeled by annotators. Consequently, the annotators must be are adequately knowledgeable about the product. Furthermore, the procedure of selecting the initial seeds is not proposed. *On the other hand, this chapter proposes a domain-free grammatical rules in Section 3.2 to construct the training set for the classifier to filter the sentences.*

3. In identifying User Sentiment State, "*based on the POS tagging, top k positive and negative terms from reviews are selected respectively as seed word lists*" [123]. The approach is questionable because part-of-speech tags do not inform the sentiment of words. Moreover, the determination of $k$ and the selected seed words may significantly affect the result. *In this chapter, the aspect sentiment analysis is performed by an attention-based LSTM (Long Short Term Memory Network) model [126] that is trained by a large corpus of similar electronic products and has been shown to perform better or comparable with the other state-of-the-art models .*

4. In obtaining the scores to cluster the Situation Facets, subjective weights are assigned to the local (containing same words) and global connection (appearing in similar reviews) scores. Furthermore, the similarity function to measure similarity between reviews to calculate global connection score is not mentioned. *In this chapter, the word vectors are used as the basis for clustering the usage contexts.* The word vectors are expected to capture the meaning of words beyond the sameness of words in phrases, because the phrases that contain same word may not refer to similar usage contexts, e.g., "playing games" and "playing music". *Also, this chapter utilizes X-means clustering to automatically obtain the number of clusters, as opposed to using k-nearest neighbor clustering [123] that requires the subjective determination of k, considering the fact that the true number of usage context clusters is unknown.*

Finally, the obtained knowledge in Ref. [62] is utilized to elicit latent needs. In Ref. [123], it is used to construct a network to explain the relations from Product and Situation Facets to User Sentiment State. In this chapter, the applications are more practical, i.e., providing visualizations (boxplots) for designers to gauge their products' positions in the market and enabling customers to filter products based on the usage contexts of their interests. It is natural for customers to express their needs in terms of usage contexts. For example, in the research about customer-oriented product design, the inputs for a mountain bike frame design originate from customers in the imprecise linguistic forms of usage purposes (e.g., speedy, free style) and contexts (e.g., rainy, rough road) [127]. Therefore, the application of this chapter should help customers naturally filters the products based on usage contexts (e.g., "video editing"), instead of based on specifications (e.g., "256 GB RAM").

Based on the similarity and differences between this chapter and Ref. [123] in particular, the two methodologies should be able to complement each other. As shown in the examples above, there are sentences that might be informative for designers but they are not identified by the methodology in Ref. [123]. On the other hand, since the methodology in Ref. [123] attempts to identify broader Situation Facets including "Time" and "Place", their methodology may produce sentences with usage contexts that are not identified by the methodology in this chapter. Since identifying usage contexts is an emerging topic in the data-driven product design domain, there are opportunities to combine, refine, and optimize the two methodologies.

### 5.2.3 Neural Network Classifier

Neural network is a machine learning method that simulates biological neurons. The network consists of interconnected nodes and it is characterized by three components, i.e., node character, network topology, and learning process [128].

The simplest neural network topology consists of an Input layer that is fully connected to an Output node [129], as shown in Figure 5.4. The node character may be explained as follows. The value of each node in Input layer, $x_i$, is multiplied by a trainable weight, $w_i$. The multiplication results are summed together and subtracted by a bias term, $\theta$, as shown in Equation 5.1. Finally, the resulting value is passed through a squashing function, $f$, to generate the output, $y$. The squashing functions that have been commonly used include sigmoid and hyperbolic tangent [130]. Sigmoid function is selected in this chapter, as it outputs a value in the range of $\left[0, 1\right]$ which may conveniently be treated as the probability of an example being a positive example.

$$y = f\left(\sum_{i=0}^{N-1} w_i x_i - \theta\right) \tag{5.1}$$

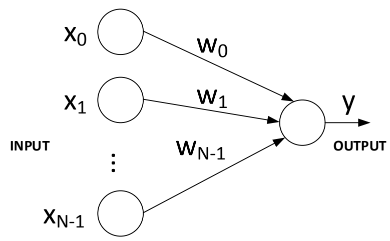



Figure 5.4: A fully-connected feed-forward neural network

As for the learning process, the network is trained by a set of labeled training data. In this chapter, each data is a sentence; in which every word in the sentence is represented by its word embedding. Therefore, if the maximum length of sentence is $M$ and the word embedding dimension is $d$, then the index $N-1$ in Figure 5.4 equals $M.d$. Each data is labeled as either 0 or 1; with 1 being a positive example. By iterating through the training data, the weights are optimized by a gradient descent approach with the objective to minimize the binary cross-entropy loss function [131].

### 5.2.4 Attention-based Long Short Term Memory Network

There are two types of neural network topology, i.e. feed-forward and recurrent [132]. As shown in Figure 5.4, a feed-forward network does not have any loop. On the other hand, a recurrent network allows loops for providing feedback. A Long Short Term Memory (LSTM) network is a recurrent network, as it takes the outputs from the previous step to be considered into the current step. The diagram in Figure 5.5(a) shows an unraveled LSTM network at steps $(t-1)$ and $t$. The relation between gates in an LSTM node is depicted in Figure 5.5(b).

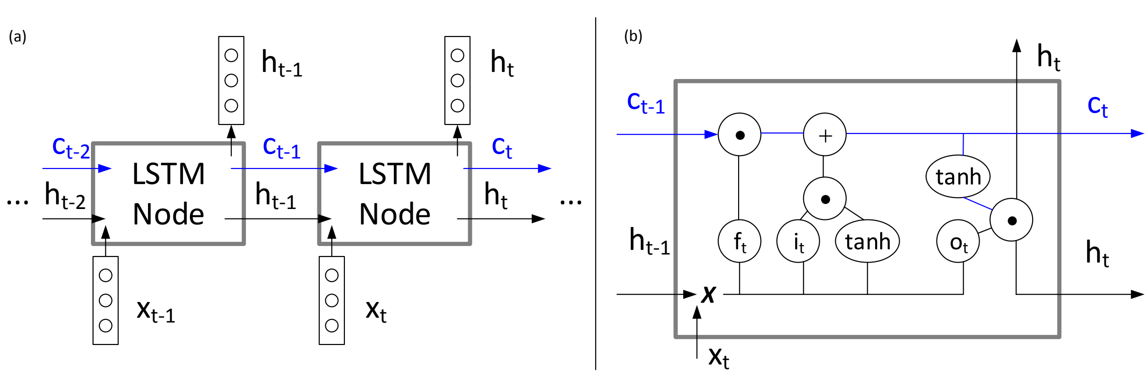



Figure 5.5: (a) A Long Short Term Memory (LSTM) neural network, (b) A diagram of gates in an LSTM node

The inputs to an LSTM node at step $t$ are the previous node's state vector, $c_{t-1}$, the previous output vector, $h_{t-1}$, and the current input vector, $x_t$. In this chapter, the input vector is a word embedding. The outputs are the current node state, $c_t$ and the current output vector, $h_t$. There are three gates in an LSTM node, i.e. Forget, Input, and Output [133]. The calculation in Forget gate is shown in Equation 5.2. In the equation, $X$ represents a vector of $[h_{t-1}, x_t]^T$, $W_f$ represents trainable weight vector, $b_f$ represents trainable bias, and $\sigma$ represents sigmoid function.

$$f_t = \sigma(W_f X + b_f) \tag{5.2}$$

The calculations in Input and Output gates use the same formula as Equation 5.2, but using their own weights $(W_i, W_o)$ and biases $(b_i, b_o)$. Based on the outputs from the gates, the node state and output vector are calculated as in Equation 5.3 and Equation 5.4, respectively.

$$c_t = f_t.(c_{t-1}) + i_t. \tanh(W_c X + b_c) \tag{5.3}$$

$$h_t = o_t. \tanh(c_t) \tag{5.4}$$

In this chapter, a method based on attention-based LSTM proposed by He et al. [126] is used to obtain aspect sentiment from a sentence. In an attention-based LSTM network, the output vectors are linearly combined into a vector, $z$, as shown in Equation 5.5. The attention, $\alpha_i$, is learned through a weight vector $W_a$ that considers both the output vector, $h_i$, and the target vector, $y$. A target vector is defined as the average of word embeddings of words in the aspect of interest. Furthermore, a softmax function is used to turn $z$ into a probability distribution over the sentiment labels, i.e. positive, neutral, and negative. The network is trained with the objective to minimize the cross-entropy loss function.

$$z = \sum_{i=1}^{n} \alpha_i h_i = \sum_{i=1}^{n} \frac{\exp(\tanh(h_i^T W_a y))}{\sum_{j=1}^{n} \exp(\tanh(h_j^T W_a y))} h_i \tag{5.5}$$

## 5.3 Methodology

The proposed methodology consists of four stages, as shown in Figure 5.6. Each stage of the methodology is discussed in the following subsections.

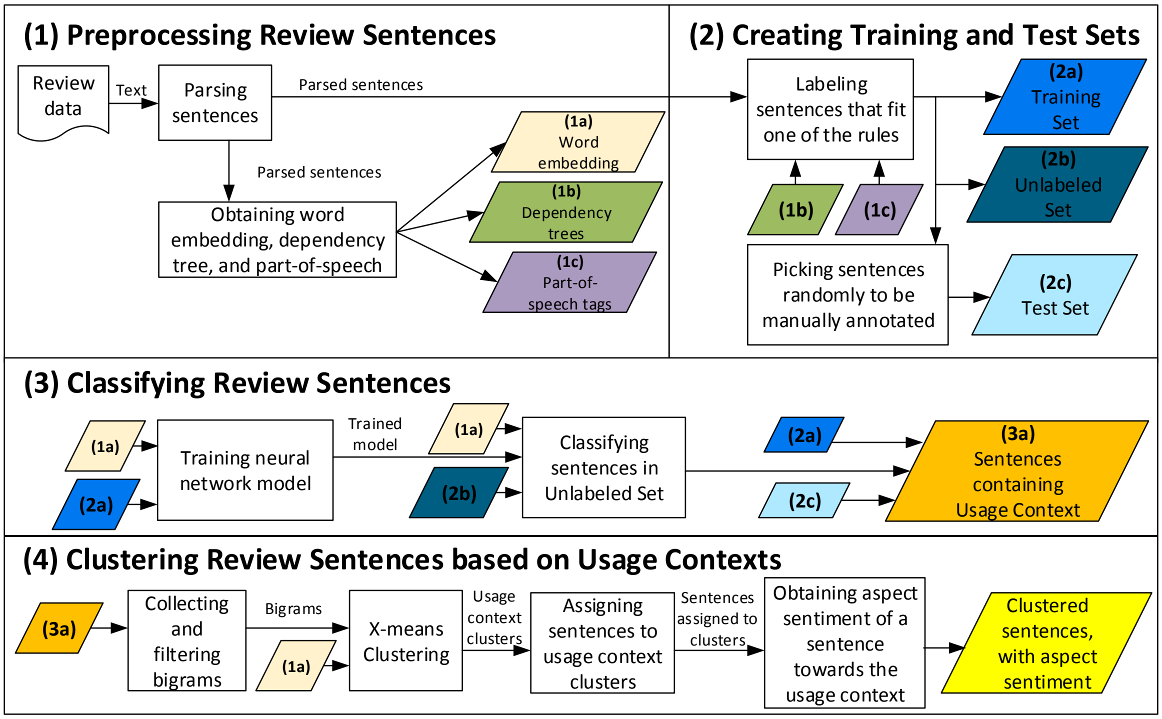



Figure 5.6: Proposed methodology to automatically identify usage contexts and cluster review sentences based on the usage contexts

### 5.3.1 Preprocessing Review Sentences

A set of customer reviews is the input to this stage. Each customer review is parsed into a set of sentences, using full stops, question marks, and exclamation marks. The sentences are subsequently parsed into dependency trees. Also, the words in the sentences are represented by word embedding and tagged by their part-of-speech tags.

A dependency tree is a representation of grammatical dependencies between words in a sentence [134]. The dependency trees become the input to create training set in Subsection 5.3.2. Word embedding are vectors of real numbers that represent words. The vectors are obtained from the technique, such as *word2vec*, that learns high quality word vectors from a data sets with a large number of words in the vocabulary [44]. The word vectors become the input to cluster words, as well as to compute the similarity between words or phrases, in Subsection 5.3.4. Part-of-speech are classes of words that have similar function with respect to the adjacent words or the affixes they take [38]. The part-of-speech tags, along with the dependency trees, become the input to create training set in Subsection 5.3.2.

### 5.3.2 Creating Training and Test Sets

A set of sentences from customer reviews, along with their dependency trees and part-of-speech tags of words, become the input to this stage. This stage creates labels for sentences, i.e., whether or not the sentences contain product usage contexts, based on several grammatical rules. Of all the sentences, there is generally a large fraction of sentences that cannot be labeled by the rules; due to the fact that the grammatical rules may not capture all grammatical variations of the sentences. Therefore, the labeled sentences become a training set to train the classifier in the next stage, which is used to classify the sentences that cannot be labeled by the grammatical rules.

The proposed grammatical rules are designed to be able to generalize to most types of product. Therefore, the rules are not designed to be highly elaborate. The example of labeled sentences that are produced by the rules are presented and discussed in Section 5. The rules are as follows:

1. *Rule 1 (For sentences that contain the word "usage")*: In the dependency tree, if the child of the word "usage" and the child's descendants include a noun or a verb, then the sentence is labeled as a positive example. Otherwise, the sentence is temporarily labeled as a negative example.

   This rule originates from the purpose of this research, i.e., identifying usage context. Therefore, it is reasonable to collect sentences that contain the word "usage". Furthermore, the child of the word "usage" with noun or verb part-of-speech is assumed to indicate a specific task or activity, e.g., "*gaming* usage". Other parts-of-speech, such as an adjective, might not provide the specificity, e.g., "*heavy* usage".

2. *Rule 2 (For sentences that contain the word "use", "used", "uses", or "using")*: If a sentence contains "use to" or "used to", it is temporarily labeled as a negative example because those phrases are frequently referred to either a routine or a past habit. If a sentence contains one of the aforementioned words, and the children of the word include the word "for" and a direct object, then it is labeled as a positive example. Otherwise, the sentence is labeled as a negative example.

   This rule is created based on the fact that a usage context may be expressed with the verb "use" and its variants. In addition, the existence of a direct object becomes an indication that a sentence is likely to contain a usage context, such as in the following example where "it" is the direct object: "right now i just use *it* for internet browsing and Pandora."

3. *Rule 3 (For sentences that do not contain the words that are queried by Rule 1 and Rule 2, but contain the word "for")*: If the child of the word "for" is tagged as a VBG (i.e., verb, gerund or present participle [135]), then the sentence is labeled as a positive example. Otherwise, the sentence is labeled

as a negative example.

This rule is created based on the fact that the preposition "for" is a function word that is used to indicate purpose, as explained by the entry in Merriam-Webster dictionary (https://www.merriam-webster.com/dictionary/for). In the example of the entry, the purpose is stated with a VBG-tagged word as well, i.e., "a grant for *studying* medicine."

A sentence that does not meet a rule's condition remains unlabeled by the rule, e.g., a sentence that has no "usage" word in it is unlabeled by Rule 1. As for Rule 1 and Rule 2, if a sentence is temporarily labeled as negative by one rule but is unlabeled by the other, then it is given a final label as negative. If a sentence is labeled as negative by both Rule 1 and Rule 2, it is also given a final label as negative. Otherwise, its final label is positive, because a sentence might contain both "usage" and "use", but only one of them meets the rule. As for Rule 3, the given label is final. The sentences with their final labels form the training set. Finally, all sentences that do not pass any rule form the unlabeled set.

As for creating the test set to evaluate the classifier's performance, a set of sentences are randomly selected and excluded from the training and unlabeled sets. In order to maintain the similar distribution to the training set, it is suggested to build half of the test set by randomly selecting sentences from the training set and build the other half by randomly selecting sentences from the unlabeled set. In order to ensure the correct label of the sentences in the test set, the sentences are manually labeled by more than one annotator. Since the product usage context in this chapter includes the tasks or applications that a user performs using the product, a question is used to guide the annotator, i.e., "Does the sentence tell the tasks or applications that a user performs using the product?" The sentences of which annotators agree on their labels form the test set.

### 5.3.3 Classifying Review Sentences

The training, test, and unlabeled sets that are created in the previous stage become the inputs to this stage. The main purpose of the classifier in this stage is to filter the sentences in the unlabeled set, which is generally larger than the other two sets, such that the sentences that contain usage contexts may be obtained without applying overly elaborate rules. Furthermore, in this chapter, the classifier performance is also used as the basis to select the hyper-parameter values in word embedding. In Section 4, several sets of hyper-parameter values are applied to the case study and the best set of hyper-parameter values is selected based on the classifier performance.

The classifier in this chapter is a one-layer neural network. The inputs are the word embedding of words in a sentence. The squashing function in the output node is a sigmoid function, such that the output is between 0 and 1. The weights are trained using the training set, in which the sentences are labeled as either positive (1) or negative (0). Once the classifier has finished the learning process, the weights may be applied to any sentence and produce a value between 0 and 1. After applying a thresholding, as explained in Subsection 5.4.2, the sentences in the unlabeled set may accordingly be labeled as either positive or negative. The sentences that are classified as negative are excluded from the next stage of the methodology.

In order to assess the performance of a classifier, the metric Area under the Receiver Operating Characteristic (AUROC) is used. The Receiver Operating Characteristic (ROC) curve is obtained by plotting the True Positive versus False Positive rates for all possible threshold values. The AUROC may then be intrepreted as, given a positive and a negative example, the probability of the classifier to output a higher prediction value for the positive example [136]. Therefore, the larger AUROC value indicates the better classifier.

### 5.3.4 Clustering Review Sentences based on Usage Contexts

The sentences that are classified as positive by the classifier become the input for this stage, along with the sentences that are labeled as positive in the training and test sets. The purpose of this stage is to cluster usage contexts, such that a sentence that contains a usage context may be clustered as well, and thus the proportion of usage contexts of a product may be obtained. Furthermore, this stage aims to reveal the sentiment in a sentence with respect to the usage context in the sentence, which is known as aspect sentiment. By obtaining the aspect sentiments, analysis on the aspect sentiment distribution among products and correlation between aspect sentiment and overall rating may be performed, as shown in Subsection 5.4.4.

In order to obtain usage contexts from the input sentences, bigrams are collected from the sentences. In this chapter, the usage contexts are assumed to be bigrams. The assumption is taken because including unigrams is expected to return a set of words that contains too much noise, i.e., words that are irrelevant to usage contexts, e.g., “wondering”. As the consequence, a one-word usage context is omitted, e.g., “writing” is omitted, but “writing paper” is included. Furthermore, the collected usage contexts should be specific enough such that they are informative for designers and useful for customers. In many cases, the specificity of a usage context may not be captured by a unigram. For example, “video” is not specific enough, as it may refer to the activity of watching video, editing video, etc.; “playing” is also not specific enough as it may refer to playing music, playing games, etc. Therefore, in order to obtain adequately specific usage contexts, this chapter assumes the usage contexts as bigrams.

The collected bigrams are subsequently clustered using X-means clustering method. It is chosen due to its ability to obtain the number of clusters automatically, by optimizing the Bayesian Information Criterion

[46]. In the context of usage contexts, it is difficult to determine the correct number of clusters of usage contexts. For example, in the case of laptops, it is highly debatable whether or not "watching movie" and "watching youtube" should be in the same cluster of usage contexts. Therefore, X-means is considered suitable for the task at this stage. Moreover, it has been shown that X-means clustering performs better than a spherical K-means clustering in the case study of laptops [137].

As for the aspect sentiment, it is obtained by applying the attention-based LSTM method proposed by He et al. [126]. The method is chosen because it is a state-of-the-art aspect sentiment analysis method, which achieves a relatively comparable or even better performance than the other recent methods, including when it is applied to a data set of laptop customer reviews from Amazon.com. The method outputs the sentiment of a sentence with respect to an aspect sentiment in three scores that sum up to 1, which correspond to positive, negative, and neutral sentiments. In this chapter, the numbers are aggregated by subtracting the negative score from the positive score. Therefore, the range of the aggregated sentiment is [-1,1].

## 5.4 Data and Results

This section starts with describing the data sets that are used to implement the proposed methodology in Section 5.3. The first subsection discusses the word embedding hyper-parameter value selection. It is followed by comparing the results from using and not using a sentence classifier. The third subsection presents the detailed results from clustering the identified usage contexts. Finally, the results from applying the aspect sentiment analysis to the usage contexts are presented.

Two data sets are used in this chapter. The Laptop data set contains 5,419 laptops from the Traditional Laptops category in Amazon.com. It also contains 218,570 customer reviews of those laptops up to December 13, 2017. The Tablet data set contains 373 tablets from BestBuy.com. It also contains 134,219 customer reviews of those tablets that are posted between November 4, 2014 and October 8, 2018. The proportions of reviews with Verified Purchase label are 85.82% and 98.66% for Laptop and Tablet data sets, respectively. Therefore, most of the reviews are expected to be authentic because they are written by customers who have been verified to purchase the products.

When a classifier is used, as proposed in the methodology shown in Figure 5.6, a training set is required to train the classifier's parameters and a test set is needed to assess the classifier's performance. Therefore, as explained in Subsection 5.3.2, the review sentences are divided into training set, unlabeled set, and test set. The number of sentences in each set for both Laptop and Tablet data sets are shown in Table 5.2.

Table 5.2: The number of sentences in each data set

| Data Set | Training Set (+) | Training Set (-) | Test Set (+) | Test Set (-) | Unlabeled Set |
|---|---|---|---|---|---|
| Laptop | 15,578 | 100,058 | 72 | 465 | 1,028,573 |
| Tablet | 5,188 | 18,520 | 51 | 166 | 318,448 |

Table 5.3: The classifier performance comparison in Laptop and Tablet data sets

| (Laptop) size | window | minCount | AUROC | | (Tablet) size | window | minCount | AUROC | |
|---|---|---|---|---|---|---|---|---|---|
| 25 | 2 | 5 | 0.8055 | | 25 | 2 | 5 | 0.7816 | |
| 25 | 2 | 10 | 0.8389 | * | 25 | 2 | 10 | 0.8110 | |
| 25 | 3 | 5 | 0.8190 | | 25 | 3 | 5 | 0.7958 | |
| 25 | 3 | 10 | 0.8158 | | 25 | 3 | 10 | 0.8280 | * |
| 50 | 2 | 5 | 0.7722 | | 50 | 2 | 5 | 0.7264 | |
| 50 | 2 | 10 | 0.7683 | | 50 | 2 | 10 | 0.7570 | |
| 50 | 3 | 5 | 0.7756 | | 50 | 3 | 5 | 0.7377 | |
| 50 | 3 | 10 | 0.7934 | | 50 | 3 | 10 | 0.7323 | |

### 5.4.1 Word Embedding Hyper-parameter Value Selection Result

The performance of a classifier is affected by the word embedding. Therefore, this subsection shows the selection of the word embedding hyper-parameter values based on the classifier performance. The hyper-parameters that are included in the experiment are the dimension of a word vector (*size*), the window size (i.e., the maximum distance between the farthest context word and the predicted word) (*window*), and the minimum freqency for a word to be included in the embedding (*minCount*). The performance of a classifier is measured by the AUROC metric.

In this chapter, the word embedding is implemented via `gensim` package in Python [74]. The word embedding becomes the input for the classifier that consists of one layer and applies a sigmoid function as the squashing function in the output layer. The classifier is implemented via `keras` package in Python. The classifier performance comparison for the selected word embedding hyper-parameter values is shown in Table 5.3. The highest AUROC value is denoted with an asterisk and the word embedding obtained from the corresponding hyper-parameter values is used in the later stages.

### 5.4.2 Sentence Classifier Result

This subsection shows the qualitative and quantitative comparisons between using a classifier to classify the review sentences, as proposed in the methodology in Section 5.3, and not using a classifier. The comparison is made in order to justify the Classifying Review Sentences stage in the proposed methodology.

After obtaining the word embedding with the best hyper-parameter values in Table 5.3, review sentences in the unlabeled set may be filtered by a classifier before entering the bigram clustering stage. First, the bigrams are collected by the CountVectorizer function of `sklearn` package in Python [138]. The collected bigrams are then refined by removing the unlikely phrases, using the phraser function of `gensim` package in Python [74]. The function is based on Pointwise Mutual Information (PMI) metric that calculates the probability of words in a phrase appearing together compared to the multiplication of the probabilities of each word appearing by itself [41]. A phrase with high PMI indicates that the phrase is likely a valid phrase.

Moreover, each word in the bigram must be either a noun or a verb and one of the words must end with "-ing". The reasoning behind that is as follows. The words that end with -ing are likely to be verbs, gerunds or present participles; which reasonably describe activities. Other than a verb, a specific bigram often contains a noun as well, e.g., typing *documents*, reading *e-books*, or even a pair of nouns that describe usage contexts, e.g., *web surfing*, *photo editing*. These filtering steps are performed in order to reduce noise in the collected bigrams and produce specific activity phrases. The final set of bigrams are clustered into usage context clusters using `pyclustering` package in Python [79]. On the other hand, without using a classifier, bigrams are immediately collected from the review sentences, refined, and clustered.

Before going into the comparisons, it is worth noting that the classifier outputs a value between 0 and 1, due to the sigmoid as the squashing function in the output layer. However, in order to classify a sentence as containing usage context or not, a binary decision is required, i.e., 0 or 1. Therefore, a thresholding process is performed. The classifier is applied to the test set and the best threshold is chosen such that the accuracy on the test set is the highest. In Laptop data set, the best threshold is obtained at 0.214, resulting in 89.01% accuracy on the test set. In Tablet data set, the best threshold is obtained at 0.323, resulting in 87.56% accuracy on the test set. The thresholding graphs are shown in Figure 5.7, in which the X-axis shows the threshold and the Y-axis shows the accuracy on the test set. The classifier is applied to the unlabeled set, with the aforementioned thresholds, and yields 25,300 positive sentences in Laptop data set and 25,556 sentences in Tablet data set. Those numbers are 2.46% and 8.02% of the sentences in the unlabeled sets of Laptop and Tablet data sets, respectively. This reduction supports the methodology to obtain relevant bigrams, as the sentences that are unlikely to contain usage contexts have been filtered out.

A qualitative comparison is made by comparing the most frequent bigrams in the clusters. In Laptop data set, the list of most frequent bigram in each cluster is as follows:

- With classifier (10 clusters): gaming rig / power saving (equally frequent), operating system, processes running, playing game, word processing, transferring files, web browsing, writing paper, video editing, watching movie.

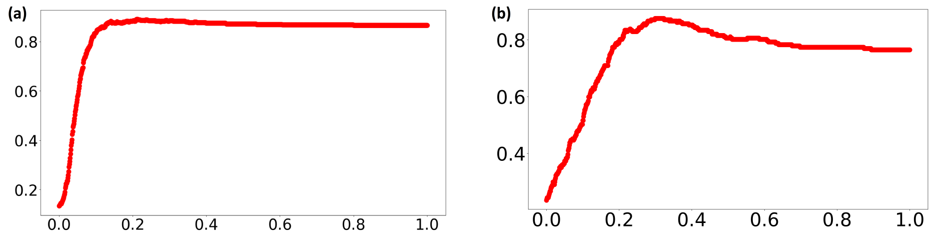



Figure 5.7: Thresholding for classifier in: (a) Laptop data set and (b) Tablet data set

Table 5.4: The comparison based on average cosine distance between with and without using a classifier

| Data Set | Within cluster (with classifier) | Within cluster (without classifier) | Between most frequent bigrams (with classifier) | Between most frequent bigrams (without classifier) |
|---|---|---|---|---|
| Laptop | 0.3806 | 0.5110 | 0.7638 | 0.9383 |
| Tablet | 0.3106 | 0.4569 | 0.7550 | 0.8194 |

- Without classifier (15 clusters): viewing angles, operating system, web browsing, video editing, stopped working*, processing speed, docking station*, learning curve, shipping label*, star rating*, processing power, cooling pad*, selling point*, transferring files, playing games.

In Tablet data set, the list of most frequent bigram in each cluster is as follows:

- With classifier (4 clusters): operating system, web browsing, watching movies, reading books.

- Without classifier (6 clusters): selling point*, operating system, photo editing, web browsing, watching movies, reading books,

It can be seen that, without using a classifier, there are frequent-but-irrelevant bigrams in the clusters of usage contexts, which are denoted by an asterisk in the list. On the other hand, the clusters that are produced from the classified sentences are represented by bigrams that are relevant to usage contexts.

A quantitative comparison is made by comparing the average cosine distance between bigrams within a cluster and between most frequent bigrams of the clusters. The smaller distance between bigrams within a cluster shows more cohesiveness of the clusters, i.e., the bigrams within a cluster have similar meaning. The smaller distance between most frequent bigrams shows that the identified frequent usage contexts are more likely to refer to the same concept, e.g., the usage contexts of laptops. The comparison is shown in Table 5.4. It can be seen that a classifier produces more cohesive clusters, as shown by the lower average cosine distance values compared to without using a classifier, as well as more similar most frequent bigrams. The result holds for both data sets.

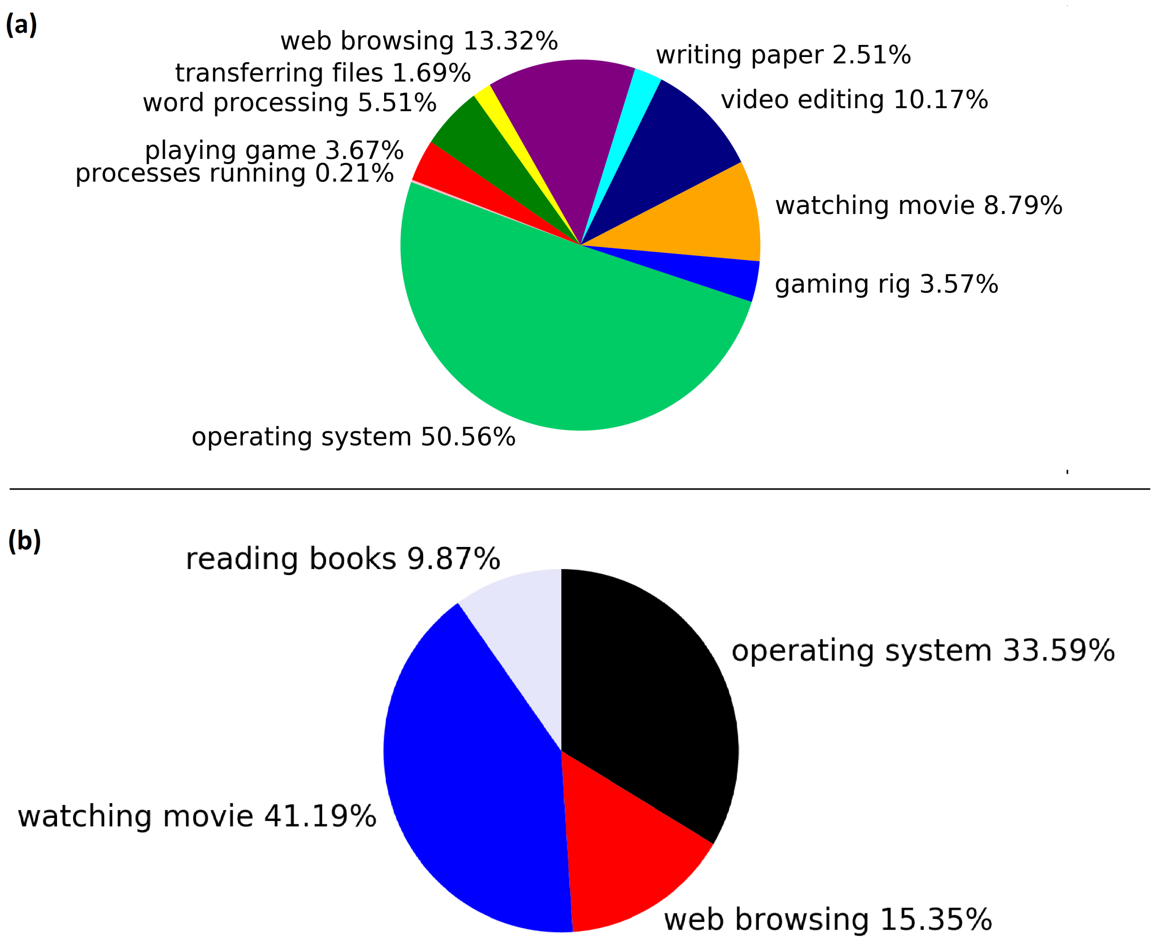



Figure 5.8: The proportion of customer reviews in each usage context cluster in: (a) Laptop data set and (b) Tablet data set

### 5.4.3 Usage Context Clustering Result

The previous subsection justifies the usage of a classifier in the proposed methodology. This subsection further observes the obtained clusters of usage contexts. Once the bigrams have been clustered into usage context clusters, the customer review sentences may be assigned to the clusters based on the usage context bigrams that are contained in the sentence. The assignment produces the charts in Figure 5.8, which show the proportions of usage contexts for both Laptop and Tablet data sets. Each fraction in in Figure 5.8 is labeled by the most frequent bigram in the cluster. It may be observed that there are fewer usage contexts identified from Tablet data set.

Table 5.5 shows a sample of frequent bigrams in each cluster in Laptop data set. The sample of frequent bigrams in Tablet data set is shown in Table 5.6. The top bigrams may be used to analyze the qualitative performance of the clustering, which will be discussed in Section 5.5.

In order to measure the quantitative performance of the clustering, the Precision metric is calculated.

Table 5.5: The sample of bigrams in each cluster in Laptop data set sorted by descending frequency

| Cluster 0 | Cluster 1 | Cluster 2 | Cluster 3 | Cluster 4 |
|---|---|---|---|---|
| operating system | gaming rig | playing game | processes running | transferring files |
| stopped working | power saving | demanding game | loading webpages | computing tasks |
| viewing angle | computing power | streaming media | handles multitasking | demanding tasks |
| stop working | computing needs | playing minecraft | — | loading pages |
| processing power | hardcore gaming | playing fallout | — | tried uninstalling |
| learning curve | engineering student | playing overwatch | — | demanding applications |
| **Cluster 5** | **Cluster 6** | **Cluster 7** | **Cluster 8** | **Cluster 9** |
| word processing | writing paper | web browsing | watching movie | video editing |
| internet surfing | reading reviews | web surfing | watching video | photo editing |
| surfing internet | writing document | internet browsing | watching netflix | streaming video |
| document processing | typing papers | checking email | watching youtube | video streaming |
| — | writing essays | browsing internet | streaming movie | document editing |
| — | reading text | surfing web | playing music | editing photo |

Table 5.6: The sample of bigrams in each cluster in Tablet data set sorted by descending frequency

| Cluster 0 | Cluster 1 | Cluster 2 | Cluster 3 |
|---|---|---|---|
| operating system | web browsing | watching movie | reading books |
| learning curve | web surfing | playing games | reading ebooks |
| processing speed | checking email | watching videos | reading magazines |
| stopped working | internet browsing | watching netflix | reading articles |
| processing power | surfing web | video editing | reading glasses |
| photo editing | surfing internet | movie watching | reading comics |

For both Laptop and Tablet data sets, 150 sentences are randomly selected to be independently labeled by two annotators. A sentence is labeled as "True" if the identified usage context is correct, but otherwise "False". The sentences that two annotators agree on the labels are collected. The Precision metric is then calculated as the ratio of the "True" sentences to the total number of sentences. The results are shown in Table 5.7. The Precision of the Laptop data set (almost 70%) is lower than Tablet data set (more than 87%) due to the fact that a laptop has more functions than a tablet, such that it has more usage contexts as well and thus it is more difficult to identify the correct usage contexts.

The performance metric Recall is not calculated here due to the difficulty in determining the true number of usage contexts. The Recall metric calculates the ratio of correctly identified usage contexts to the number of true usage contexts. The difficulty lies in the fact that the true number of usage contexts depends on the granularity of the contexts. For example, in the sentence in Figure 5.1, "loading apps", "(loading) webpages", and "opening documents" may be considered as three different types of usage contexts. However, "loading apps" and "opening documents" may reasonably be merged into one context, i.e., "loading apps", because "opening documents" may be interpreted as loading an app to open a document as well. The more complicated difficulty is when one activity may be considered as a subset of the other. For example, in the

Table 5.7: Precision of the identified usage contexts

| Data Set | True | False | Precision (%) |
|---|---|---|---|
| Laptop | 85 | 37 | 69.67 |
| Tablet | 118 | 17 | 87.41 |

context of customer review in Figure 5.1, "writing papers" is a subset of "doing research". It is not clear whether or not those two usage contexts should be counted separately. Suppose the methodology identifies "writing papers" but not "doing research", it is unclear whether it should be counted as a miss. In fact, doing research is an unspecific term that may include various activities such as web browsing, watching video, running simulation, etc.; such that it is arguably acceptable to either identify it as a usage context or not.

### 5.4.4 Aspect Sentiment Analysis Result

Aspect sentiment analysis is performed for the most frequent bigram in each usage context cluster, as the representation of the cluster. First, the aspect sentiment analysis is used to show the distribution of sentiment towards a particular usage context among all products in the data set. Therefore, product A, for example, may compare its relative position to product B based on average customer sentiment with respect to "video editing" usage context. Furthermore, product A may obtain its relative position among all products in the data set with respect to that usage context. For Laptop and Tablet data sets, the aspect sentiment distributions are shown in Figure 5.9 and Figure 5.10, respectively. It may be observed that the customers in Tablet data sets are generally more positive towards the product in all usage contexts.

Moreover, in order to examine whether or not there is a strong linear correlation between aspect sentiment towards a particular usage context and the overall rating, the boxplots in Figure 5.11 and Figure 5.12 are created. For each usage context, a plot that consists of 4 boxplots are created. The X-axis corresponds to the ranges of aspect sentiment of [-1, 0.5), [-0.5, 0), [0, 0.5), and [0.5, 1]. The Y-axis is the overall rating of a product. The interpretation of the plots may be made as follows. For example, in Figure 5.11, for the usage context "playing game", the laptops that have average sentiment towards that context in the range of [-1,0.5) are the laptops whose overall rating median is around 4. There is an outlier laptop in that group, whose overall rating is below 2. The boxplots, along with the correlation coefficient values, demonstrate that there is a weak to moderate positive correlation between aspect sentiment and overall rating for most of the usage contexts. In other word, it is shown here that the higher overall rating of a product does not strongly correlate to a higher sentiment towards a particular usage context of the product.

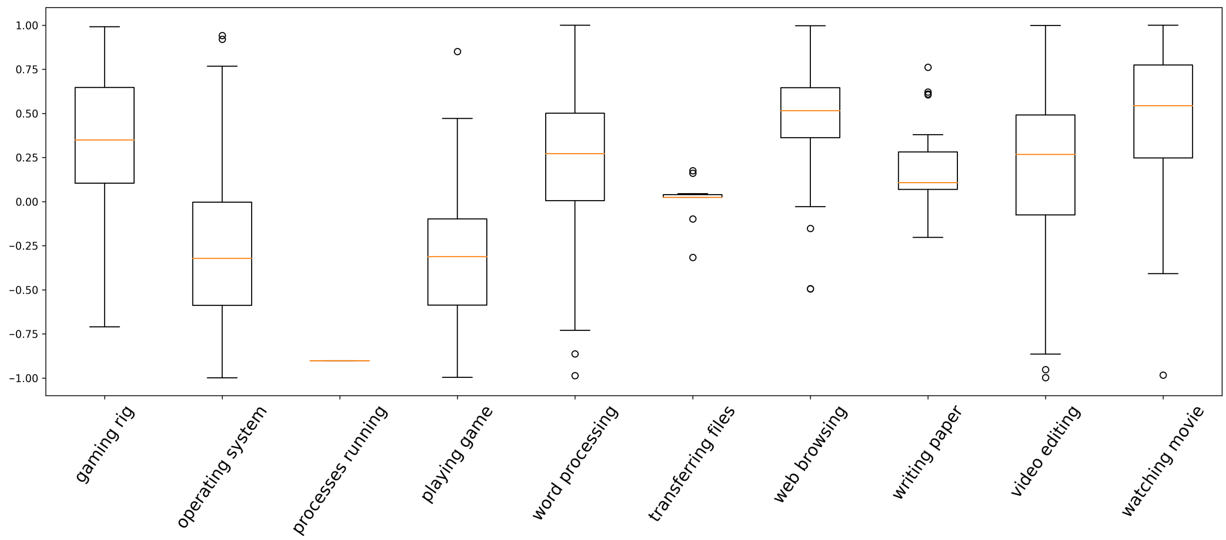



Figure 5.9: Boxplots of the aspect sentiment related to each usage context in Laptop data set

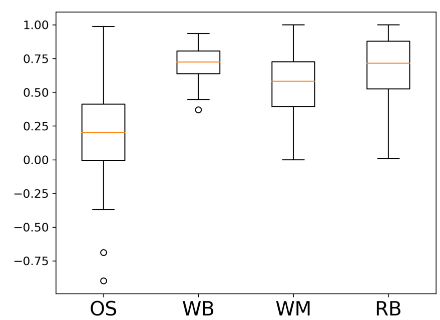



Figure 5.10: Boxplots of the aspect sentiment related to each usage context in Tablet data set, which OS = operating system, WB = web browsing, WM = watching movie, and RB = reading books

## 5.5 Discussion

This section first discusses the proposed methodology's performance based on the results in Section 5.4. It is followed by the subsections related to the contributions of the methodology for customers and product designers.

### 5.5.1 Methodology's Performance

The grammatical rules that are used to label sentences generally produce correct labels. The examples of positive and negative sentences that are produced by the rules in Laptop data set are presented as follows, in which the sentences are retained in their original writings including the grammatical and typographical errors:

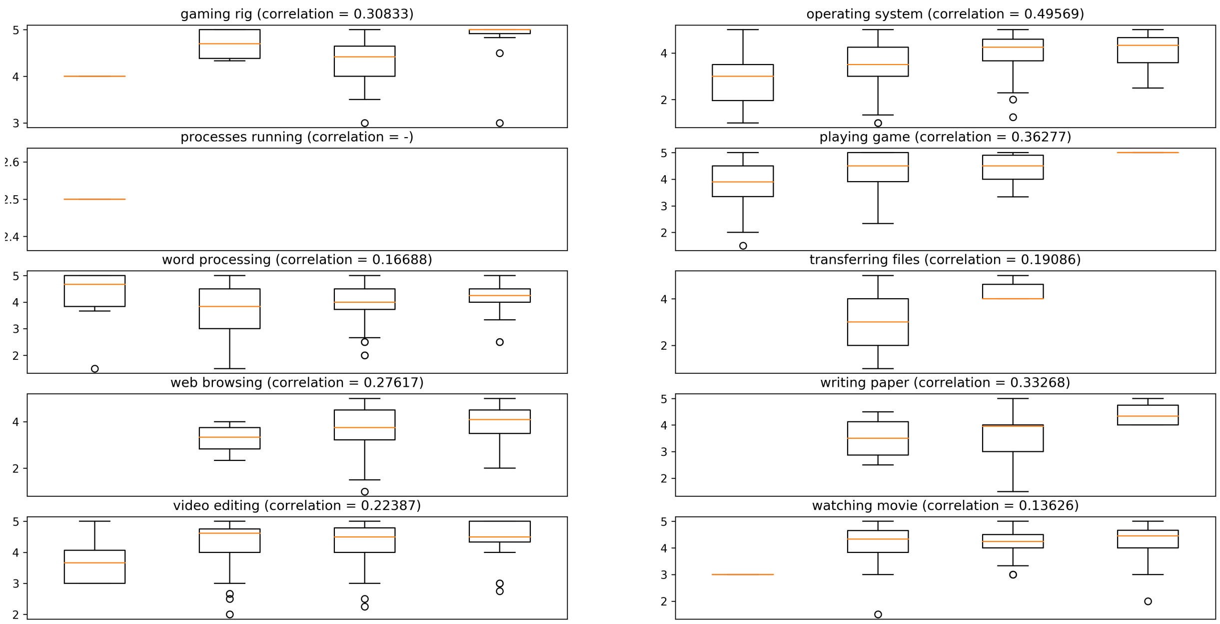



Figure 5.11: Boxplots of the aspect sentiment related to each use case in Laptop data set

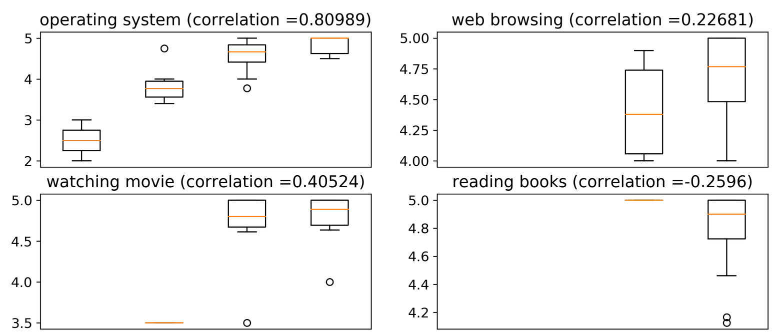



Figure 5.12: Boxplots of the aspect sentiment related to each use case in Tablet data set

- *Rule 1 (Positive)*: "it's slim and lightweight , not too fast , not recomended formulti tasking or complex software , but is ok for *everyday* **usage** *like web browsing , email , word processor , etc*"
- *Rule 1 (Negative, because the child of the word "usage", i.e., "moderate", is neither a noun nor a verb)*: "battery life is perfect , it lasts 12 hours as stated in the description and up to 10 hours on moderate **usage**"
- *Rule 2 (Positive)*: "if you plan to **use** the *laptop* **for** more than *browsing* and *watching movies* then you might consider it a waste of $ 250"
- *Rule 2 (Negative, because the children of the word "use" do not include a direct object and the word "for")*: "it automatically slects which one to **use** based on what you are doing to conserve battery power"

- *Rule 3 (Positive)*: "the non - glare finish is much better than the glossy displays gracing many other notebooks , and the ips display has dramatic superiority **for** *viewing angle accuracy*"
- *Rule 3 (Negative, because the part-of-speech tag of the child of the word "for" is not VBG)*: "i was lucky enough to get mine **for** 1000 as it was mis - marked at the base exchange where i bought it"

The rules also produce false positive examples, such as "i **use** it pretty much everyday **for** *extended periods* of *time* and only charge a few times a week " (Rule 2), and false negative examples, such as "i bought it mainly **for** work , they have desk - tops there but we have to log - in with our clock # and they can watch your every move and put you on the corporate i / t watchlist if you transgress , omg google images , he looked up what" (Rule 3). In the false positive example, "extended periods (of) time" is identified as a usage context. Meanwhile, in the false negative example, "work" is not identified as a usage context. A set of more elaborate rules may produce fewer false examples. Nevertheless, the rules in the proposed methodology are intentionally designed to be not too detailed, such that the rules may generalize to other product domains and there is as little subjective input as possible in the methodology.

As for the qualitative performance of the clustering, Table 5.5 displays a reasonable result in clustering "gaming rig" and "hardcore gaming" together with "computing power" in Cluster 1, which indicates the needs for computing power. Meanwhile, there is another cluster, i.e., Cluster 2, that groups usage contexts that are similar to "playing game", which may be interpreted as requiring less computing power. This result indicates that the proposed method is able to capture the meaning behind the bigrams, instead of simply capturing bigrams that contain the same word. This argument is further supported by the separation of "playing music" in Cluster 8 and "playing game" in Cluster 2, although both bigrams contain the word "playing". The proposed methodology is not totally accurate, obviously, because there are some bigrams that seem to fit better in another cluster, e.g., "streaming media" in Cluster 2 is intuitively more compatible with the terms in Cluster 8.

When a product can be considered as a subset of the other in terms of functionality, such as tablet to laptop, the methodology obtains less number of usage contexts as well. It can be seen that there are 4 clusters of usage contexts in Tablet data set in Table 5.6, compared to 12 in Table 5.5 for Laptop data set. This result qualitatively justifies the ability of the methodology in obtaining usage contexts from customer reviews.

Aside from the limitations that arise from the inacurracy of machine learning and Natural Language Processing tools, the proposed methodology is unable to weigh the usage contexts that are mentioned by one customer. The weights should capture the finer level of the importance of different usage contexts for a customer. For example, a customer might comment positively on the wide viewing angle for a laptop,

Table 5.8: Example of two laptops with their average aspect sentiments for two usage contexts

| Product | Context: watching movie | Context: video editing |
|---|---|---|
| Laptop 7a (B005CWJB5G) | 0.8283 | 0.0661 |
| Laptop 7b (B007474DSM) | 0.8239 | 0.5699 |

but the customer does not find it important, because the customer's main usage context is streaming music. Therefore, it would be appropriate to weigh the customer's positiveness accordingly.

### 5.5.2 Contribution for Customers

The proposed methodology may benefit customers in a way as follows. Suppose a customer compares two laptops as shown in Table 5.8, along with their average aspect sentiments with respect to "watching movie" and "video editing" usage contexts. The methodology allows customers to notice that, while both laptops have similar average sentiment for "watching movie", Laptop 7b has a significantly higher average sentiment for "video editing". Therefore, if the customer considers both usage contexts as important, the comparison may cause Laptop 7b to be preferable for the customer. Under the current filtering options in Amazon.com and BestBuy.com, it is difficult for customer to filter and compare laptops by these criteria.

The sample of review sentences for both laptops in Table 5.8 with respect to both usage contexts are shown in Table 5.9. The review sentences are presented to qualitatively justify the sentiment scores. In Table 5.9, it may be observed that both laptops receive sentences with positive sentiment towards "watching movie" usage context. For "video editing" usage context, Laptop 7a has been mostly described as being capable for light video editing. On the other hand, Laptop 7b has been positively described as being suitable for video editing, except for the fourth sentence that complains about the nonexistence of a set of numeric keys on the keyboard. Therefore, as shown in Table 5.8, Laptop 7b has a higher average sentiment value than Laptop 7a for "video editing" usage context.

### 5.5.3 Contribution for Designers

For the designers, Figure 5.9 and Figure 5.10 may be used to identify the opportunity in the market. In the case of laptops, the improved products may be targeted for the usage contexts of playing game and operating system. Those are the usage contexts for which most of the laptops are perceived negatively by the customers. In the case of tablets, there is also an opportunity to improve the operating system in order to stand out from the competitors.

Moreover, in a more detailed level, designers may examine the extracted sentences from the customer

Table 5.9: The sample of review sentences for two laptops in Table 5.8 with respect to the corresponding usage contexts

| | | |
|---|---|---|
| **7a** | • battery life is amazing , i get 4 hours of youtube watching or **watching movies** on a plane<br>• in addition , i'll admit that i kind of use it as a "portable dvd player" **watching movies** in bed when i'm too lazy to head over to my mac pro | • i only do very minor music and **video editing**<br>• **video editing** is functional as well and would likely work for most casual users , but massive projects simply wouldn't be possible on this machine for a multitude of reasons ranging from storage space to video card , screen size , processor etc<br>• if you have a main computer , the air is a very good addition but if you only need an all in one computer like me to take with you , store data , watch videos , do picture and **video editing** and plan to use for a very long time , then the macbook pro is a better choice and for me is more sturdy<br>• if you were a graphic designer or video editor , this might not have enough power for you , but i've done a bit of **video editing** in imovie on this , and it worked really well , and didn't slow down at all |
| **7b** | • i travel a lot and **watching movies** / tv shows while on flights with the retina display is amazing<br>• why i , personally , chose the retina mbp : - i use my machine for **watching movies** , work in grad school , internet , and a potpurri of other things | • i mainly use it for **video editing** , photo editing , household management and other personal tasks<br>• if you use your mac for picture editing or for **video editing** its a good use of your money<br>• those who don't require this performance might want to look at other macs , but if you run graphic - intense programs , do **video editing** , watch a lot of media via your computer , apple has really delivered<br>• the problem is that this laptop has a great screen , runs fast and therefore should be great for **video editing** on the road , but i use the 10 key portion of a keyboard when video editing , and this laptop does not have one<br>• with intel's latest turboboost - capable ivy bridge processors , up to 16gb of sdram and a potential of 768gb of the fastest flash storage available , the retina macbook pro can easily accomplish simple tasks such as video playback , as well as the more complex - such as hardware - intensive **video editing** |

Table 5.10: The example of review sentences from the products with the most positive and negative average aspect sentiment for a particular usage context

| Product | Sentence | Average |
|---|---|---|
| Laptop 9a (B0030INLSW) | "i'm liking windows 7 , and the computer comes with ms works which gives you as much as most need for **writing papers** or doing spreadsheets"<br>"overall i'm satisfied : i have a huge screen for studying and **writing papers** , the keyboard is a great design allowing for comfortable typing with responsive keys , and appropriately clicky buttons" | 0.76262 |
| Laptop 9b<br><br>(B01K1IO3QW) | "it works fine for me , someone just using it for college and **writing papers** but i wouldn't buy it again"<br>"my only intention was to use this computer for **writing papers** and doing research and in the week that i had the computer i was not able to do either" | -0.20224 |

reviews with respect to a particular usage context. For example, the products that have the highest and the lowest average aspect sentiment with respect to the usage context of "writing papers" are shown in Table 5.10, along with the corresponding review sentences and the average aspect sentiments.

Taking Laptop 9b as an example, the designers of Laptop 9b may want to improve their product, since it currently has the lowest sentiment with respect to "writing papers" usage context compared to all other laptops in the data set. The improvement becomes essential if Laptop 9b targets customers who frequently write papers on laptops. While the sentence may not offer the complete problem description by itself, the designer of Laptop 9b may carefully examine the entire review from this particular customer as shown in Figure 5.1. The review reveals that the customer experiences the need to reinstall the operating system, although in fact the laptop has been equipped with Windows 10 Home. Also, the customer perceives the

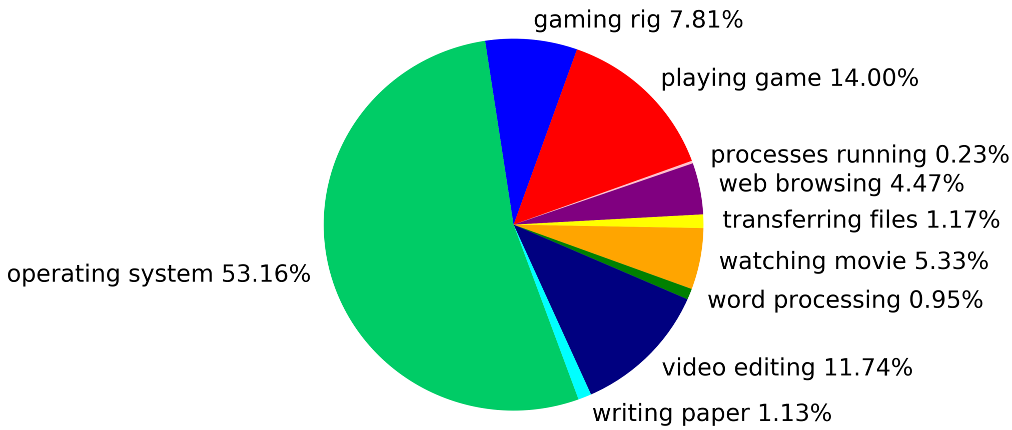



Figure 5.13: The proportion of customer reviews in each usage context cluster in gaming laptops

laptop as extremely slow in performing basic functions. The result has therefore significantly narrowed down the number of customer reviews that a product designer needs to focus on.

Addressing the importance of obtaining actual, as opposed to assumed, usage context [113], a pie chart is created in order to show the usage contexts of gaming laptops, i.e., the laptops that contain the words "Gamer", "Gaming", "Alienware", and "MSI" in their names. The latter two terms are the brands of gaming laptops. The chart in Figure 5.13 shows that gaming laptops obviously have larger proportions in the usage contexts of "gaming rig" and "playing game" compared those in the overall Laptop data set in Figure 5.8. It may be seen that the proportions of several other usage contexts are not negligible. Therefore, the gaming laptop designers should not assume that customers only use the laptops for gaming purpose, especially since there have been negative sentiments towards these other usage contexts. Furthermore, the negative sentiment may include a suggestion for improvement, as shown by the following sentence: "- there is *no dedicated pgup / pgdn key* on the razer , kind of annoying during **web browsing**", which is written towards "web browsing" usage context. The improvement might be beneficial in order to attract people who frequently use gaming laptops for web browsing as well. By noticing the usage contexts that might have not been previously considered as important, designers might formulate design improvements in order to attract either the targeted or new customers.

## 5.6 Conclusion

A data-driven methodology has been proposed to automatically identify product usage contexts from online customer reviews. The theoretical contributions of this chapter are: (1) proposing grammatical rules, which are not specific to a particular product domain, to create a data set for training a sentence classifier (Subsection 5.3.2), (2) proposing a sentence classifier to obtain sentences that contain usage contexts (Subsection 5.3.3), and (3) identifying usage contexts from customer review sentences, as well as obtaining their corresponding aspect sentiments; even when the sentences may not contain either product-feature or sentiment words.

When the identified product usage contexts are complemented with aspect sentiment analysis, the interpretation of the results may be beneficial in several ways. For designers, the results may be used to evaluate the position of a product with respect to its competitors in different usage contexts, which enables the identification of product improvements and market opportunities. For customers, the results provides opportunity to filter products based on the sentiment towards their prioritized usage contexts. It has also been shown that the overall rating is not strongly correlated with the sentiment towards individual usage contexts.

To address the strength of utilizing online reviews for identifying product usage context, the main benefits are the amount of data and its availability. Data may be obtained, analyzed, and interpreted in a time period that is faster than the required time to design, obtain approval, and conduct a survey-based method. Consequently, companies may be able to make faster decisions in many aspects, e.g., changing the advertisement strategy after learning about customers' usage contexts, deciding to improve the next generation product's performance in a particular usage context, etc. On the other hand, sentences in online reviews may not always provide detailed usage contexts. In contrast to survey-based methods, there are vague usage contexts that cannot be easily clarified or verified. For example, when a review states "doing research", it is hard to clarify the type of research activities. Also, when a review states that "photo editing is slow", it is hard to verify whether the laptop is actually incapable of performing the task or, for example, the user has not installed the software correctly. Moreover, survey-based method may provide a higher granularity of the result. For example, the survey-based methods might reveal that people who complain about "photo editing" are mostly graphic designers.

Chapter 6

# Conclusion and Future Work

The research in this dissertation has shown the promising role of publicly available data, such as online customer reviews, for decision making in engineering design. The publicly available data may complement or, to some extent, replace the conventional survey-based data collection methods. Accordingly, this dissertation proposes methodologies to utilize the sheer amount of customer review data specifically for supporting product design decisions. This section summarizes and concludes the research, as well as presenting the potential improvements for the future work.

## 6.1 Conclusion

In Chapter 3, a methodology is presented to automatically identify product features that are significantly related to sales rank. The product features are identified by applying word embedding and clustering techniques. The product features are related to the sentiment words based on dependency-tree-based rules. The relation between product features, their corresponding sentiments, and sales rank is obtained from a linear regression model. The methodology identifies, among all product features, a subset of product features and their corresponding sentiments that is significantly related to sales rank. For product designers, the result suggests to focus on the reviews that contain the discussion of that set of product features. By focusing on those reviews, instead of reading a massive amount of reviews, designers may learn customer sentiment and feedback about the product and thus plan accordingly to improve the product on the next generation.

In Chapter 4, a methodology is proposed to assist designers in estimating demands of a product that has a specific combination of attribute values. In the engineering design domain, the demand for a product is often modeled by a choice model. The construction of choice models relies on the choice sets, which are frequently obtained by conducting survey-based methods. In the proposed methodology, a novel approach of utilizing online customer reviews is proposed to build the choice sets, which are subsequently used to construct the choice models. The approach utilizes the clustering techniques to cluster the products and customers. Products are clustered based on the values of their attributes. Customers are clustered based on the product features and sentiments that they discuss in their reviews. The methodology shows that the information from both product and customer clusters results in choice models with higher predictive ability compared to the models that use random choice sets. For product designers, the quick construction of choice models provides an opportunity to improve the current product by developing and evaluating alternative product concepts in the earliest time possible compared to waiting for the results from conducting survey-based methods.

In Chapter 5, a methodology is designed to automatically identify the activities and tasks that customers perform on a product, i.e., the usage contexts of a product. The usage contexts are important because they greatly affect customers' experience towards the product. The usage contexts are identified from the customer reviews by collecting and clustering bigrams from the review sentences that pass a classifier. The sentence classifier is trained by sentences that are labeled by dependency-tree-based rules. Furthermore, an aspect sentiment analysis is applied to capture customer sentiment towards the usage contexts. The methodology is able to identify usage contexts from the online customer reviews in the case study with good precision. For product designers, the identification is useful in several ways. First, designers may verify if the intended usage contexts are perceived positively by customers. Second, designers may assess whether or not there is a discrepancy between the intended contexts and the actual contexts of usage. Finally, designers may use the results in order to identify usage contexts that are previously not considered as important or are currently perceived with negative sentiments by most customers in the market. Referring to Kano's model, these usage contexts may become possible product improvements that excite customers.

Since the main data of this research are online customer reviews, it is worth addressing the issue of inauthentic reviews here. In particular, with respect to the three main chapters in this dissertation, the implications of the inauthentic reviews being possibly included in the data are as follows.

1. In Chapter 3, the work is identifying the product-feature and sentiment words that are related to sales rank. In this case, regardless of the authenticity of the reviews, the identified product-feature and sentiment words from the reviews are found to be significantly related to sales rank. Therefore, in other word, the results in this chapter should not be affected by the authenticity of the reviews.

2. In Chapter 4, the work is constructing customer choice sets using online customer reviews. Specifically, the choice sets are constructed based on product and customer clusters. Customer clusters are formed based on customer reviews. In this case, the inauthentic reviews may affect the clustering, such that it consequently affects the probability sampling to construct the choice sets. As the result, the parameters of the constructed choice models may not be as accurate as they should be. In order to minimize the effects from the inauthentic reviews, the customer reviews that are used in this chapter are the ones

that have been labeled as “Verified Purchase” by Amazon.com. Therefore, the reviewers have been verified to actually purchase the product that they review and the risk of having inauthentic reviews has been reduced.

3. In Chapter 5, the work is identifying usage contexts of a product using online customer reviews. In this case, the existence of inauthentic reviews may affect the identified usage contexts, as well as the aspect sentiment towards those contexts. However, in the presented case studies, the proportions of inauthentic reviews are expected to be small because majority of the reviews are labeled as “Verified Purchase”, i.e., the website (either Amazon or BestBuy) has verified that the reviewer has actually purchased the product in review. Therefore, the results should be expected to reflect the actual usage contexts and aspect sentiments of the real customers.

## 6.2 Future Work

Based on the works in the three main chapters in this dissertation, i.e., Chapter 3, 4, and 5; there are several interesting future works that may be pursued as follows.

1. In order to improve the accuracy of interpreting customer reviews, a better word embedding may be achieved by applying word sense disambiguation [38] to words that have multiple meanings. In the case of wearable technology products, for example, the word “charge” may mean either refilling a battery by passing a current through it or the name of a product variant from Fitbit. Also, an improved method may be required to determine more accurate connections between a pair of product-feature and sentiment words based on the relations in a dependency tree.

   On the data aspect, utilizing the actual market share data would be ideal. In this research, sales rank data is used as the proxy of the market share. Since ranking is ordinal, the difference between two ranking numbers does not always translate to the same difference in market share. For example, the market share difference between the products ranked $1^{st}$ and $2^{st}$ may not be the same between the products ranked $101^{st}$ and $102^{nd}$.

2. In order to obtain a more comprehensive representation of a customer for building choice models, other types of online self-presentation may be considered to characterize a customer, e.g., past purchase history, review history, and reviewer rank. Moreover, acquiring the actual choice sets of the customers who write customer reviews would be very valuable for performance evaluation. In this research, the choice models based on the proposed methodology have higher predictive ability than the models based

on random choice sets. However, its relative performance compared to the models based on the actual choice sets is not known.

Furthermore, different types of logit models may be used to extend the methodology for wider types of products. For example, nested logit might be appropriate for products with a nested structure, such as in-car DVD players.

3. In order to assist product designers in designing product with attractive attributes that excite customers –which are often difficult to foresee [5]; the main challenge would be extending the product usage context identification to identifying extraordinary usage contexts. Extraordinary usage contexts are important, since they are related to lead users, i.e., the customers that use a product in an extraordinary context such that they reveal latent needs that are crucial for product innovation [62]. The challenge lies in the fact that the frequency of these extraordinary contexts are generally very low. Therefore, it is challenging to identify them among a massive number of irrelevant terms that appear with low frequency as well. Other forms of words (unigram, trigrams, etc.) may also be considered as the basis to identify usage contexts. The challenge with unigrams would be inferring the activity from a sentence that implicitly mentions the activity, e.g., "good for spending time with *videos* on youtube every day" –in which the activity is more likely to be watching video, instead of editing video. The usage context identification may also become the basis to construct a cross-product choice set in choice modeling, since a choice set may be formed by different types of items that serve the same usage intent [94]. Therefore, the proposed method may contribute to construct, for example, the set of devices (both laptops and tablets) that are compatible for the usage context of "web browsing".

# References


[1] Murphy, J., and Roser, M., 2018. Internet, (accessed August 13, 2018). See also URL https://ourworldindata.org/internet.

[2] eMarketer Report, 2018. Worldwide retail and ecommerce sales: emarketer's updated forecast and new mcommerce estimates for 2016-2021, (accessed August 13, 2018). See also URL https://www.emarketer.com/Report/Worldwide-Retail-Ecommerce-Sales-eMarketers-Updated-Forecast-New-Mcommerce-Estimates-20162021/2002182.

[3] Ulrich, K. T., and Eppinger, S. D., 2004. *Product Design and Development*, 3rd ed. McGraw-Hill/Irwin.

[4] McAuley, J., and Leskovec, J., 2013. "Hidden factors and hidden topics: Understanding rating dimensions with review text". In Proceedings of the 7th ACM Conference on Recommender Systems, RecSys '13, ACM, pp. 165–172. See also URL http://doi.acm.org/10.1145/2507157.2507163.

[5] Jin, J., Liu, Y., Ji, P., and Kwong, C. K., 2018. "Review on recent advances in information mining from big consumer opinion data for product design". *Journal of Computing and Information Science in Engineering,* **19**(1), pp. 0108011–01080119. See also URL http://dx.doi.org/10.1115/1.4041087.

[6] Arndt, J., 1967. "Role of product-related conversations in the diffusion of a new product". *Journal of Marketing Research,* **4**(3), pp. 291–295. See also URL http://www.jstor.org/stable/3149462.

[7] Godes, D., and Mayzlin, D., 2004. "Using online conversations to study word-of-mouth communication". *Marketing Science,* **23**(4), pp. 545–560. See also URL https://doi.org/10.1287/mksc.1040.0071.

[8] Dellarocas, C., 2003. "The digitization of word of mouth: Promise and challenges of online feedback mechanisms". *Management Science,* **49**(10), pp. 1407–1424. See also URL https://doi.org/10.1287/mnsc.49.10.1407.17308.

[9] Hennig-Thurau, T., Gwinner, K. P., Walsh, G., and Gremler, D. D., 2004. "Electronic word-of-mouth via consumer-opinion platforms: What motivates consumers to articulate themselves on the internet?". *Journal of Interactive Marketing,* **18**(1), pp. 38 – 52. See also URL https://doi.org/10.1002/dir.10073.

[10] Cao, Q., Duan, W., and Gan, Q., 2011. "Exploring determinants of voting for the helpfulness of online user reviews: A text mining approach". *Decision Support Systems,* **50**(2), pp. 511 – 521. See also URL http://www.sciencedirect.com/science/article/pii/S0167923610001909.

[11] Cheung, C. M., and Lee, M. K., 2012. "What drives consumers to spread electronic word of mouth in online consumer-opinion platforms". *Decision Support Systems,* **53**(1), pp. 218 – 225. See also URL http://www.sciencedirect.com/science/article/pii/S0167923612000413.

[12] eMarketer, 2009. Online reviews sway shoppers, (accessed July 30, 2018). See also URL https://www.emarketer.com/Article/Online-Reviews-Sway-Shoppers/1006404.

[13] Sun, M., 2012. "How does the variance of product ratings matter?". *Management Science,* **58**(4), pp. 696–707. See also URL http://dx.doi.org/10.1287/mnsc.1110.1458.

[14] Arpita, A., and Saurabh, B., 2016. “Online review helpfulness: Role of qualitative factors”. *Psychology & Marketing,* **33**(11), pp. 1006–1017. See also URL `https://onlinelibrary.wiley.com/doi/abs/10.1002/mar.20934`.

[15] López-López, I., and Parra, J. F., 2016. “Is a most helpful ewom review really helpful? the impact of conflicting aggregate valence and consumer’s goals on product attitude”. *Internet Research,* **26**(4), pp. 827–844. See also URL `https://search.proquest.com/docview/1802127512?accountid=14553`.

[16] Barbara, B., and M., S. R., 2001. “Internet forums as influential sources of consumer information”. *Journal of Interactive Marketing,* **15**(3), pp. 31–40. See also URL `https://onlinelibrary.wiley.com/doi/abs/10.1002/dir.1014`.

[17] Goldsmith, R. E., and Horowitz, D., 2006. “Measuring motivations for online opinion seeking”. *Journal of Interactive Advertising,* **6**(2), pp. 2–14. See also URL `https://doi.org/10.1080/15252019.2006.10722114`.

[18] Jensen, M. L., Averbeck, J. M., Zhang, Z., and Wright, K. B., 2013. “Credibility of anonymous online product reviews: A language expectancy perspective”. *Journal of Management Information Systems,* **30**(1), pp. 293–324. See also URL `https://doi.org/10.2753/MIS0742-1222300109`.

[19] Cardoso, E. F., Silva, R. M., and Almeida, T. A., 2018. “Towards automatic filtering of fake reviews”. *Neurocomputing,* **309**, pp. 106 – 116. See also URL `https://doi.org/10.1016/j.neucom.2018.04.074`.

[20] Jindal, N., and Liu, B., 2008. “Opinion spam and analysis”. In Proceedings of the 2008 International Conference on Web Search and Data Mining, WSDM ’08, ACM, pp. 219–230. See also URL `http://doi.acm.org/10.1145/1341531.1341560`.

[21] Lim, E.-P., Nguyen, V.-A., Jindal, N., Liu, B., and Lauw, H. W., 2010. “Detecting product review spammers using rating behaviors”. In Proceedings of the 19th ACM International Conference on Information and Knowledge Management, CIKM ’10, ACM, pp. 939–948. See also URL `http://doi.acm.org/10.1145/1871437.1871557`.

[22] Zhang, D., Zhou, L., Kehoe, J. L., and Kilic, I. Y., 2016. “What online reviewer behaviors really matter? effects of verbal and nonverbal behaviors on detection of fake online reviews”. *Journal of Management Information Systems,* **33**(2), pp. 456–481. See also URL `https://doi.org/10.1080/07421222.2016.1205907`.

[23] Mukherjee, A., Liu, B., Wang, J., Glance, N., and Jindal, N., 2011. “Detecting group review spam”. In Proceedings of the 20th International Conference Companion on World Wide Web, WWW ’11, ACM, pp. 93–94. See also URL `http://doi.acm.org/10.1145/1963192.1963240`.

[24] Mukherjee, A., Liu, B., and Glance, N., 2012. “Spotting fake reviewer groups in consumer reviews”. In Proceedings of the 21st International Conference on World Wide Web, WWW ’12, ACM, pp. 191–200. See also URL `http://doi.acm.org/10.1145/2187836.2187863`.

[25] Zhou, L., and Zhang, D., 2008. “Following linguistic footprints: Automatic deception detection in online communication”. *Commun. ACM,* **51**(9), Sept., pp. 119–122. See also URL `http://doi.acm.org/10.1145/1378727.1389972`.

[26] Mukherjee, A., Kumar, A., Liu, B., Wang, J., Hsu, M., Castellanos, M., and Ghosh, R., 2013. “Spotting opinion spammers using behavioral footprints”. In Proceedings of the 19th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD ’13, ACM, pp. 632–640. See also URL `http://doi.acm.org/10.1145/2487575.2487580`.

[27] Blei, D. M., Ng, A. Y., and Jordan, M. I., 2003. “Latent dirichlet allocation”. *J. Mach. Learn. Res.,* **3**, Mar., pp. 993–1022. See also URL `http://dl.acm.org/citation.cfm?id=944919.944937`.

[28] Zhuang, M., Cui, G., and Peng, L., 2018. "Manufactured opinions: The effect of manipulating online product reviews". *Journal of Business Research,* **87**, pp. 24 – 35. See also URL `https://doi.org/10.1016/j.jbusres.2018.02.016`.

[29] Lim, S., and Tucker, C. S., 2016. "A bayesian sampling method for product feature extraction from large-scale textual data". *Journal of Mechanical Design,* **138**(6), pp. 0614031–0614039. See also URL `http://dx.doi.org/10.1115/1.4033238`.

[30] Shi, F., Chen, L., Han, J., and Childs, P., 2017. "A data-driven text mining and semantic network analysis for design information retrieval". *Journal of Mechanical Design,* **139**(11), pp. 1114021–11140214. See also URL `http://dx.doi.org/10.1115/1.4037649`.

[31] Zhang, R., and Tran, T., 2011. "An information gain-based approach for recommending useful product reviews". *Knowledge and Information Systems,* **26**(3), Mar, pp. 419–434. See also URL `https://doi.org/10.1007/s10115-010-0287-y`.

[32] Zheng, X., Zhu, S., and Lin, Z., 2013. "Capturing the essence of word-of-mouth for social commerce: Assessing the quality of online e-commerce reviews by a semi-supervised approach". *Decision Support Systems,* **56**, pp. 211 – 222. See also URL `https://doi.org/10.1016/j.dss.2013.06.002`.

[33] Qi, J., Zhang, Z., Jeon, S., and Zhou, Y., 2016. "Mining customer requirements from online reviews: A product improvement perspective". *Information and Management,* **53**(8), pp. 951 – 963. Big Data Commerce.

[34] Zhang, Z., Liu, L., Wei, W., Tao, F., Li, T., and Liu, A., 2017. "A systematic function recommendation process for data-driven product and service design". *Journal of Mechanical Design,* **139**(11), pp. 1114041–11140414. See also URL `http://dx.doi.org/10.1115/1.4037610`.

[35] Suryadi, D., and Kim, H., 2018. "A systematic methodology based on word embedding for identifying the relation between online customer reviews and sales rank". *Journal of Mechanical Design,* **140**(12), pp. 1214031–12140312. See also URL `http://dx.doi.org/10.1115/1.4040913`.

[36] Chaklader, R., and Parkinson, M., 2017. "Data-driven sizing specification utilizing consumer text reviews". *Journal of Mechanical Design - Transactions of the ASME,* **139**(11), 11, pp. 1114061–1114067. See also URL `https://doi.org/10.1115/1.4037476`.

[37] Chiu, M.-C., and Lin, K.-Z., 2018. "Utilizing text mining and kansei engineering to support data-driven design automation at conceptual design stage". *Advanced Engineering Informatics,* **38**, pp. 826 – 839. See also URL `https://doi.org/10.1016/j.aei.2018.11.002`.

[38] Jurafsky, D., and Martin, J. H., 2009. *Speech and Language Processing*, $2^{nd}$ ed. Pearson Education, Inc., Upper Saddle River, New Jersey.

[39] Levy, O., and Goldberg, Y., 2014. "Dependency-based word embeddings". In Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers), Association for Computational Linguistics, pp. 302–308. See also URL `http://www.aclweb.org/anthology/P/P14/P14-2050`.

[40] Somprasertsri, G., 2010. "Mining Feature-Opinion in Online Customer Reviews for Opinion Summarization". *Journal of Universal Computer Science,* **16**(6), pp. 938–955.

[41] Mikolov, T., Sutskever, I., Chen, K., Corrado, G., and Dean, J., 2013. "Distributed representations of words and phrases and their compositionality". *CoRR,* **abs/1310.4546**. See also URL `http://arxiv.org/abs/1310.4546`.

[42] Zhang, D., Xu, H., Su, Z., and Xu, Y., 2015. "Chinese comments sentiment classification based on word2vec and SVMperf". *Expert Systems with Applications,* **42**(4), pp. 1857–1863. See also URL `http://dx.doi.org/10.1016/j.eswa.2014.09.0116`.

[43] Rong, X., 2014. "word2vec parameter learning explained". *CoRR,* **abs/1411.2738**. See also URL `http://arxiv.org/abs/1411.2738`.

[44] Mikolov, T., Chen, K., Corrado, G., and Dean, J., 2013. "Efficient estimation of word representations in vector space". *CoRR,* **abs/1301.3781**. See also URL `https://arxiv.org/abs/1301.3781`.

[45] Berkhin, P., 2006. *A Survey of Clustering Data Mining Techniques.* Springer Berlin Heidelberg, Berlin, Heidelberg, pp. 25–71. See also URL `https://doi.org/10.1007/3-540-28349-8_2`.

[46] Pelleg, D., and Moore, A. W., 2000. "X-means: Extending k-means with efficient estimation of the number of clusters". In Proceedings of the Seventeenth International Conference on Machine Learning, ICML '00, Morgan Kaufmann Publishers Inc., pp. 727–734. See also URL `http://portal.acm.org/citation.cfm?id=657808`.

[47] Kotler, P., and Keller, K. L., 2006. *Marketing Management*, $12^{th}$ ed. Pearson Education, Inc., Upper Saddle River, New Jersey, ch. 6, pp. 184–199.

[48] Decker, R., and Trusov, M., 2010. "Estimating aggregate consumer preferences from online product reviews". *International Journal of Research in Marketing,* **27**(4), pp. 293 – 307.

[49] Chevalier, J. A., and Mayzlin, D., 2006. "The effect of word of mouth on sales: Online book reviews". *Journal of Marketing Research,* **43**(3), pp. 345–354. See also URL `https://doi.org/10.1509/jmkr.43.3.345`.

[50] Jindal, N., and Liu, B., 2007. "Review spam detection". In Proceedings of the 16th International Conference on World Wide Web, WWW '07, ACM, pp. 1189–1190. See also URL `http://doi.acm.org/10.1145/1242572.1242759`.

[51] Guo, H., Zhu, H., Guo, Z., Zhang, X., and Su, Z., 2009. "Product feature categorization with multilevel latent semantic association". In Proceedings of the 18th ACM Conference on Information and Knowledge Management, CIKM '09, ACM, pp. 1087–1096. See also URL `http://doi.acm.org/10.1145/1645953.1646091`.

[52] Suryadi, D., and Kim, H. M., 2017. "Identifying sentiment-dependent product features from online reviews". In Design Computing and Cognition '16, J. S. Gero, ed., Springer International Publishing, pp. 685–701. See also URL `https://doi.org/10.1007/978-3-319-44989-0`.

[53] Suryadi, D., and Kim, H., 2016. "Identifying the relations between product features and sales rank from online reviews". In ASME 2016 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference, p. V02AT03A015. See also URL `http://dx.doi.org/10.1115/DETC2016-60481`.

[54] Chong, A. Y. L., Chng, E., Liu, M. J., and Li, B., 2017. "Predicting consumer product demands via big data: the roles of online promotional marketing and online reviews". *International Journal of Production Research,* **55**(17), pp. 5142–5156. See also URL `https://doi.org/10.1080/00207543.2015.1066519`.

[55] Quan, X., Wenyin, L., and Dou, W., 2011. "Longitudinal sales responses with online reviews". In Proceedings of the 11th IEEE International Conference on Data Mining Workshops, ICDMW, IEEE, pp. 103–108. See also URL `https://doi.org/10.1109/ICDMW.2011.115`.

[56] Archak, N., Ghose, A., and Ipeirotis, P. G., 2007. "Show me the money!: Deriving the pricing power of product features by mining consumer reviews". In Proceedings of the 13th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '07, ACM, pp. 56–65. See also URL `http://doi.acm.org/10.1145/1281192.1281202`.

[57] Kobayashi, N., Inui, K., and Matsumoto, Y., 2007. "Extracting aspect-evaluation and aspect-of relations in opinion mining". In Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning, pp. 1065–1074. See also URL `http://www.aclweb.org/anthology/D07-1114`.

[58] Netzer, O., Feldman, R., Goldenberg, J., and Fresko, M., 2012. "Mine your own business: Market-structure surveillance through text mining". *Marketing Science,* **31**(3), pp. 521–543. See also URL `https://doi.org/10.1287/mksc.1120.0713`.

[59] Hu, M., and Liu, B., 2004. "Mining opinion features in customer reviews". In Proceedings of the 19th National Conference on Artifical Intelligence, AAAI'04, AAAI Press, pp. 755–760. See also URL `http://dl.acm.org/citation.cfm?id=1597148.1597269`.

[60] Hu, M., and Liu, B., 2004. "Mining and summarizing customer reviews". In Proceedings of the Tenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '04, ACM, pp. 168–177. See also URL `http://doi.acm.org/10.1145/1014052.1014073`.

[61] Archak, N., Ghose, A., and Ipeirotis, P. G., 2011. "Deriving the Pricing Power of Product Features by Mining Consumer Reviews". *Management Science,* **57**(8), pp. 1485–1509. See also URL `http://dx.doi.org/10.1287/mnsc.1110.1370`.

[62] Zhou, F., Jiao, R. J., and Linsey, J. S., 2015. "Latent customer needs elicitation by use case analogical reasoning from sentiment analysis of online product reviews". *Journal of Mechanical Design,* **137**, pp. 0714011 – 07140112. See also URL `http://dx.doi.org/10.1115/1.4030159`.

[63] Wei, C.-P., Chen, Y.-M., Yang, C.-S., and Yang, C. C., 2010. "Understanding what concerns consumers: a semantic approach to product feature extraction from consumer reviews". *Information Systems and e-Business Management,* **8**(2), pp. 149 – 167. See also URL `https://doi.org/10.1007/s10257-009-0113-9`.

[64] Abulaish, M., Jahiruddin, Doja, M. N., and Ahmad, T., 2009. "Feature and opinion mining for customer review summarization". In Pattern Recognition and Machine Intelligence, S. Chaudhury, S. Mitra, C. A. Murthy, P. S. Sastry, and S. K. Pal, eds., Springer Berlin Heidelberg, pp. 219–224.

[65] Somprasertsri, G., and Lalitrojwong, P., 2008. "Automatic product feature extraction from online product reviews using maximum entropy with lexical and syntactic features". In 2008 IEEE International Conference on Information Reuse and Integration, pp. 250–255.

[66] Jin, W., Ho, H. H., and Srihari, R. K., 2009. "Opinionminer: A novel machine learning system for web opinion mining and extraction". In Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '09, ACM, pp. 1195–1204. See also URL `http://doi.acm.org/10.1145/1557019.1557148`.

[67] Ma, B., Zhang, D., Yan, Z., and Kim, T., 2013. "An LDA and Synonym Lexicon Based Approach To Product Feature Extraction From Online Consumer Product Reviews". *Journal of Electronic Commerce,* **14**(4), pp. 304–314. See also URL `http://www.jecr.org/node/345`.

[68] Mei, Q., Ling, X., Wondra, M., Su, H., and Zhai, C., 2007. "Topic sentiment mixture: Modeling facets and opinions in weblogs". In Proceedings of the 16th International Conference on World Wide Web, WWW '07, ACM, pp. 171–180.

[69] Zhai, Z., Liu, B., Xu, H., and Jia, P., 2011. "Clustering product features for opinion mining". In Proceedings of the Fourth ACM International Conference on Web Search and Data Mining, WSDM '11, ACM, pp. 347–354. See also URL `http://doi.acm.org/10.1145/1935826.1935884`.

[70] Kim, S.-M., and Hovy, E., 2006. "Extracting opinions, opinion holders, and topics expressed in online news media text". In Proceedings of the Workshop on Sentiment and Subjectivity in Text, SST '06, Association for Computational Linguistics, pp. 1–8. See also URL `http://dl.acm.org/citation.cfm?id=1654641.1654642`.

[71] Cambria, E., Poria, S., Bajpai, R., and Schuller, B. W., 2016. "Senticnet 4: A semantic resource for sentiment analysis based on conceptual primitives". In COLING 2016, 26th International Conference on Computational Linguistics, Proceedings of the Conference: Technical Papers, December 11-16, 2016, Osaka, Japan, pp. 2666–2677. See also URL `http://aclweb.org/anthology/C/C16/C16-1251.pdf`.

[72] Bird, S., Klein, E., and Loper, E., 2009. *Natural Language Processing with Python*, 1st ed. O'Reilly Media, Inc.

[73] Manning, C. D., Surdeanu, M., Bauer, J., Finkel, J., Bethard, S. J., and McClosky, D., 2014. "The Stanford CoreNLP natural language processing toolkit". In Association for Computational Linguistics (ACL) System Demonstrations, pp. 55–60.

[74] Řehůřek, R., and Sojka, P., 2010. "Software Framework for Topic Modelling with Large Corpora". In Proceedings of the LREC 2010 Workshop on New Challenges for NLP Frameworks, ELRA, pp. 45–50. See also URL `http://is.muni.cz/publication/884893/en`.

[75] Quan, C., and Ren, F., 2014. "Unsupervised product feature extraction for feature-oriented opinion determination". *Information Sciences,* **272**, pp. 16–28. See also URL `http://dx.doi.org/10.1016/j.ins.2014.02.063`.

[76] Cambria, E., Hussain, A., Havasi, C., and Eckl, C., 2009. "Affectivespace: Blending common sense and affective knowledge to perform emotive reasoning". In In WOMSA09, Seville, pp. 32–41.

[77] Forman, C., Ghose, A., and Wiesenfeld, B., 2008. "Examining the relationship between reviews and sales: The role of reviewer identity disclosure in electronic markets". *Information Systems Research,* **19**(3), pp. 291–313. See also URL `http://pubsonline.informs.org/doi/abs/10.1287/isre.1080.0193`.

[78] Miller, A., 2002. *Subset Selection in Regression*, $2^{nd}$ ed. Chapman & Hall/CRC, Boca Raton, FL, USA.

[79] Novikov, A., 2018. annoviko/pyclustering: pyclustering 0.8.1 release, May. See also URL `https://doi.org/10.5281/zenodo.1254845`.

[80] Li, H., and Azarm, S., 2000. "Product design selection under uncertainty and with competitive advantage". *Journal of Mechanical Design,* **122**(4), pp. 411–418. See also URL `http://dx.doi.org/10.1115/1.1311788`.

[81] Kumar, D., Chen, W., and Simpson, T. W., 2009. "A market-driven approach to product family design". *International Journal of Production Research,* **47**(1), pp. 71–104. See also URL `https://doi.org/10.1080/00207540701393171`.

[82] Michalek, J., Ebbes, P., Adigzel, F., Feinberg, F., and Papalambros, P., 2011. "Enhancing marketing with engineering: Optimal product line design for heterogeneous markets". *International Journal of Research in Marketing,* **28**, 03, pp. 1–12. See also URL `https://doi.org/10.1016/j.ijresmar.2010.08.001`.

[83] He, L., Chen, W., Hoyle, C., and Yannou, B., 2012. "Choice modeling for usage context-based design". *Journal of Mechanical Design,* **134**(3), pp. 0310071–03100711. See also URL `http://dx.doi.org/10.1115/1.4005860`.

[84] Morrow, W. R., Long, M., and MacDonald, E. F., 2014. "Market-system design optimization with consider-then-choose models". *Journal of Mechanical Design,* **136**(3), pp. 0310031–03100313. See also URL `http://dx.doi.org/10.1115/1.4026094`.

[85] Train, K. E., 2003. *Discrete Choice Methods with Simulation.* Cambridge University Press, Cambridge, U.K., MA.

[86] Wang, M., and Chen, W., 2015. "A data-driven network analysis approach to predicting customer choice sets for choice modeling in engineering design". *Journal of Mechanical Design,* **137**(7), pp. 0714101–07141011. See also URL `http://dx.doi.org/10.1115/1.4030160`.

[87] Chen, W., Wassenaar, H. J., and Hoyle, C., 2013. *Decision-Based Design: Integrating Consumer Preferences into Engineering Design.* Springer-Verlag, London.

[88] McFadden, D., 1978. "Modeling the choice of residential location". *Transportation Research Record,* **673**, pp. 72–77. See also URL http://onlinepubs.trb.org/Onlinepubs/trr/1978/673/673-012.pdf.

[89] Kang, S., 2018. "Warehouse location choice: A case study in Los Angeles, CA". *Journal of Transport Geography.* See also URL https://doi.org/10.1016/j.jtrangeo.2018.08.007.

[90] Ioannides, Y. M., and Zabel, J. E., 2008. "Interactions, neighborhood selection and housing demand". *Journal of Urban Economics,* **63**(1), pp. 229 – 252. See also URL https://doi.org/10.1016/j.jue.2007.01.010.

[91] Peters, T., Adamowicz, W. L., and Boxall, P. C., 1995. "Influence of choice set considerations in modeling the benefits from improved water quality". *Water Resources Research,* **31**(7), pp. 1781–1787. See also URL https://doi.org/10.1029/95WR00975.

[92] Valencia-Romero, A., and Lugo, J. E., 2017. "An immersive virtual discrete choice experiment for elicitation of product aesthetics using gestalt principles". *Design Science,* **3**, p. e11. See also URL http://dx.doi.org/10.1017/dsj.2017.12.

[93] Gensch, D. H., 1987. "A two-stage disaggregate attribute choice model". *Marketing Science,* **6**(3), pp. 223–239. See also URL https://doi.org/10.1287/mksc.6.3.223.

[94] Shocker, A. D., Ben-Akiva, M., Boccara, B., and Nedungadi, P., 1991. "Consideration set influences on consumer decision-making and choice: Issues, models, and suggestions". *Marketing Letters,* **2**(3), pp. 181–197. See also URL http://www.jstor.org/stable/40216215.

[95] Newman, M. E. J., and Girvan, M., 2004. "Finding and evaluating community structure in networks". *Phys. Rev. E,* **69**, Feb, p. 026113. See also URL https://link.aps.org/doi/10.1103/PhysRevE.69.026113.

[96] Dominick, J. R., 1999. "Who do you think you are? Personal home pages and self-presentation on the world wide web". *Journalism & Mass Communication Quarterly,* **76**(4), pp. 646–658. See also URL https://doi.org/10.1177/107769909907600403.

[97] Gosling, S. D., Augustine, A. A., Vazire, S., Holtzman, N., and Gaddis, S., 2011. "Manifestations of personality in online social networks: Self-reported facebook-related behaviors and observable profile information". *Cyberpsychology, Behavior and Social Networking,* **14**(9), pp. 483–488. See also URL http://doi.org/10.1089/cyber.2010.0087.

[98] Nosko, A., Wood, E., and Molema, S., 2010. "All about me: Disclosure in online social networking profiles: The case of facebook". *Computers in Human Behavior,* **26**(3), pp. 406 – 418. See also URL http://www.sciencedirect.com/science/article/pii/S0747563209001836.

[99] Li, J., and Chignell, M., 2010. "Birds of a feather: How personality influences blog writing and reading". *International Journal of Human-Computer Studies,* **68**(9), pp. 589 – 602. See also URL http://www.sciencedirect.com/science/article/pii/S1071581910000522.

[100] Wagner, C., Asur, S., and Hailpern, J., 2013. "Religious politicians and creative photographers: Automatic user categorization in twitter". In Proceedings of the 2013 International Conference on Social Computing, IEEE, pp. 303 – 310. See also URL https://doi.org/10.1109/SocialCom.2013.49.

[101] Marriott, T. C., and Buchanan, T., 2014. "The true self online: Personality correlates of preference for self-expression online, and observer ratings of personality online and offline". *Computers in Human Behavior,* **32**, pp. 171 – 177. See also URL http://www.sciencedirect.com/science/article/pii/S0747563213004354.

[102] Bigi, B., 2003. "Using kullback-leibler distance for text categorization". In Advances in Information Retrieval, F. Sebastiani, ed., Springer Berlin Heidelberg, pp. 305–319. See also URL https://doi.org/10.1007/3-540-36618-0_22.

[103] Hauser, J. R., and Wernerfelt, B., 1990. "An evaluation cost model of consideration sets". *Journal of Consumer Research,* **16**(4), pp. 393–408. See also URL http://www.jstor.org/stable/2489451.

[104] Lovreglio, R., Fonzone, A., and dellOlio, L., 2016. "A mixed logit model for predicting exit choice during building evacuations". *Transportation Research Part A: Policy and Practice,* **92**, pp. 59 – 75. See also URL http://www.sciencedirect.com/science/article/pii/S096585641630550X.

[105] Kim, D., and Park, B.-J. R., 2017. "The moderating role of context in the effects of choice attributes on hotel choice: A discrete choice experiment". *Tourism Management,* **63**, pp. 439 – 451. See also URL http://www.sciencedirect.com/science/article/pii/S0261517717301656.

[106] de Jong, G., Kouwenhoven, M., Ruijs, K., van Houwe, P., and Borremans, D., 2016. "A time-period choice model for road freight transport in flanders based on stated preference data". *Transportation Research Part E: Logistics and Transportation Review,* **86**, pp. 20 – 31. See also URL http://www.sciencedirect.com/science/article/pii/S136655451500229X.

[107] Dias, F. F., Lavieri, P. S., Garikapati, V. M., Astroza, S., Pendyala, R. M., and Bhat, C. R., 2017. "A behavioral choice model of the use of car-sharing and ride-sourcing services". *Transportation,* **44**(6), Nov, pp. 1307–1323. See also URL https://doi.org/10.1007/s11116-017-9797-8.

[108] Brathwaite, T., and Walker, J. L., 2018. "Asymmetric, closed-form, finite-parameter models of multinomial choice". *Journal of Choice Modelling.* See also URL http://www.sciencedirect.com/science/article/pii/S1755534516301117.

[109] Jones, E., Oliphant, T., Peterson, P., et al., 2001–. SciPy: Open source scientific tools for Python. See also URL http://www.scipy.org/.

[110] Koren, Y., Bell, R., and Volinsky, C., 2009. "Matrix factorization techniques for recommender systems". *Computer,* **42**(8), Aug, pp. 30–37. See also URL https://doi.org/10.1109/MC.2009.263.

[111] Green, M. G., Rajan, P. K. P., and Wood, K. L., 2004. "Product usage context: Improving customer needs gathering and design target setting". In ASME 2004 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference, pp. 393 – 403. See also URL http://dx.doi.org/10.1115/DETC2004-57498.

[112] Green, M. G., Tan, J., Linsey, J. S., Seepersad, C. C., and Wood, K. L., 2005. "Effects of product usage context on consumer product preferences". In ASME 2005 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference, pp. 171 – 185. See also URL http://dx.doi.org/10.1115/DETC2005-85438.

[113] Kanis, H., 1998. "Usage centred research for everyday product design". *Applied Ergonomics,* **29**(1), pp. 75 – 82. See also URL https://doi.org/10.1016/S0003-6870(97)00028-8.

[114] Belk, R. W., 1975. "Situational variables and consumer behavior". *Journal of Consumer Research,* **2**(3), pp. 157–164. See also URL http://www.jstor.org/stable/2489050.

[115] Ram, S., and Jung, H.-S., 1991. "How product usage influences consumer satisfaction". *Marketing Letters,* **2**(4), Nov, pp. 403–411. See also URL https://doi.org/10.1007/BF00664226.

[116] Ratneshwar, S., and Shocker, A. D., 1991. "Substitution in use and the role of usage context in product category structures". *Journal of Marketing Research,* **28**(3), pp. 281–295. See also URL http://www.jstor.org/stable/3172864.

[117] Green, M. G., Linsey, J. S., Seepersad, C. C., Wood, K. L., and Jensen, D. J., 2006. "Frontier design: A product usage context method". In ASME 2006 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference, pp. 99 – 113. See also URL http://dx.doi.org/10.1115/DETC2006-99608.

[118] Lim, S., and Tucker, C., 2017. "Mitigating online product rating biases through the discovery of optimistic, pessimistic, and realistic reviewers". *Journal of Mechanical Design - Transactions of the ASME,* **139**(11), 11, pp. 1114091–11140911. See also URL `https://doi.org/10.1115/1.4037612`.

[119] Jiang, H., Kwong, C. K., and Yung, K. L., 2017. "Predicting future importance of product features based on online customer reviews". *Journal of Mechanical Design,* **139**(11), pp. 1114131–11141310. See also URL `http://dx.doi.org/10.1115/1.4037348`.

[120] LaFleur, R. S., 1992. "Principal engineering design questions". *Research in Engineering Design,* **4**(2), Jun, pp. 89–100. See also URL `https://doi.org/10.1007/BF01580147`.

[121] Ram, S., and Jung, H.-S., 1990. "The conceptualization and measurement of product usage". *Journal of the Academy of Marketing Science,* **18**(1), Dec, pp. 67–76. See also URL `https://doi.org/10.1007/BF02729763`.

[122] Ghosh, D., Olewnik, A., and Lewis, K., 2017. "Application of feature-learning methods toward product usage context identification and comfort prediction". *Journal of Computing and Information Science in Engineering,* **18**(1), pp. 0110041–01100410. See also URL `http://dx.doi.org/10.1115/1.4037435`.

[123] Yang, B., Liu, Y., Liang, Y., and Tang, M., 2019. "Exploiting user experience from online customer reviews for product design". *International Journal of Information Management,* **46**, pp. 173 – 186. See also URL `https://doi.org/10.1016/j.ijinfomgt.2018.12.006`.

[124] Liu, B., and Zhang, L., 2012. *A Survey of Opinion Mining and Sentiment Analysis*. Springer US, Boston, MA, pp. 415–463. See also URL `https://doi.org/10.1007/978-1-4614-3223-4_13`.

[125] Cambria, E., Poria, S., Gelbukh, A., and Thelwall, M., 2017. "Sentiment analysis is a big suitcase". *IEEE Intelligent Systems,* **32**(6), November, pp. 74–80. See also URL `https://doi.org/10.1109/MIS.2017.4531228`.

[126] He, R., Lee, W. S., Ng, H. T., and Dahlmeier, D., 2018. "Exploiting document knowledge for aspect-level sentiment classification". In Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers), Association for Computational Linguistics, pp. 579–585. See also URL `http://aclweb.org/anthology/P18-2092`.

[127] Gologlu, C., and Mizrak, C., 2011. "An integrated fuzzy logic approach to customer-oriented product design". *Journal of Engineering Design,* **22**(2), pp. 113–127.

[128] Zou, J., Han, Y., and So, S.-S., 2009. *Overview of Artificial Neural Networks*. Humana Press, Totowa, NJ, pp. 14–22. See also URL `https://doi.org/10.1007/978-1-60327-101-1_2`.

[129] Lippmann, R., 1987. "An introduction to computing with neural nets". *IEEE ASSP Magazine,* **4**(2), Apr, pp. 4–22. See also URL `https://doi.org/10.1109/MASSP.1987.1165576`.

[130] Refenes, A. N., Azema-Barac, M., Chen, L., and Karoussos, S. A., 1993. "Currency exchange rate prediction and neural network design strategies". *Neural Computing & Applications,* **1**(1), Mar, pp. 46–58. See also URL `https://doi.org/10.1007/BF01411374`.

[131] Drozdzal, M., Vorontsov, E., Chartrand, G., Kadoury, S., and Pal, C., 2016. "The importance of skip connections in biomedical image segmentation". In Deep Learning and Data Labeling for Medical Applications, G. Carneiro, D. Mateus, L. Peter, A. Bradley, J. M. R. S. Tavares, V. Belagiannis, J. P. Papa, J. C. Nascimento, M. Loog, Z. Lu, J. S. Cardoso, and J. Cornebise, eds., Springer International Publishing, pp. 179–187. See also URL `https://doi.org/10.1007/978-3-319-46976-8_19`.

[132] Jain, A. K., Mao, J., and Mohiuddin, K. M., 1996. "Artificial neural networks: a tutorial". *Computer,* **29**(3), March, pp. 31–44. See also URL `https://doi.org/10.1109/2.485891`.

[133] Wang, Y., Huang, M., Zhu, X., and Zhao, L., 2016. "Attention-based lstm for aspect-level sentiment classification". In Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, Association for Computational Linguistics, pp. 606–615. See also URL `http://aclweb.org/anthology/D16-1058`.

[134] Culotta, A., and Sorensen, J., 2004. "Dependency tree kernels for relation extraction". In Proceedings of the 42Nd Annual Meeting on Association for Computational Linguistics, ACL '04, Association for Computational Linguistics. See also URL `https://doi.org/10.3115/1218955.1219009`.

[135] Santorini, B., 1990. Part-Of-Speech tagging guidelines for the Penn Treebank project (3rd revision). Tech. rep.

[136] Pepe, M. S., 2000. "Receiver operating characteristic methodology". *Journal of the American Statistical Association,* **95**(449), pp. 308–311. See also URL `https://doi.org/10.1080/01621459.2000.10473930`.

[137] Suryadi, D., and Kim, H., 2019. "Automatic identification of product usage contexts from online customer reviews". In Proceedings of the International Conference on Engineering Design, ICED (In press).

[138] Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., and Duchesnay, E., 2011. "Scikit-learn: Machine learning in python". *J. Mach. Learn. Res.,* **12**, Nov., pp. 2825–2830. See also URL `http://dl.acm.org/citation.cfm?id=1953048.2078195`.